



UNIVERSITÀ DI PISA

Facoltà di Economia
Facoltà di Scienze, Matematiche, Fisiche e Naturali

Corso di laurea magistrale in Informatica per l'economia
e per l'azienda (Business Informatics)

TESI DI LAUREA
PROGETTAZIONE E REALIZZAZIONE DI UN DATA WAREHOUSE
SUI TICKET DI SERVIZIO E SULLE PERFORMANCE DEL
PERSONALE

RELATORE

Prof. Salvatore RUGGIERI

Candidato

Gianfranco DAGA

ANNO ACCADEMICO 2012-13

SOMMARIO

Il lavoro descritto in questo documento riguarda la completa automazione della reportistica legata al sistema informativo relativo alla collaborazione che la multinazionale *Accenture* svolge con *Velux*, una compagnia danese di edilizia, tramite la creazione di un *data warehouse*. Il sistema preesistente prevedeva l'inserimento manuale da parte del personale Accenture, operante in diverse postazioni sparse tra l'Europa e l'Asia, di una consistente quantità di dati e la creazione manuale di alcuni report, fattore che si è rivelato determinante, nel tempo, per lo sviluppo di errori e incongruenze. La reportistica è lo strumento analitico del quale i dirigenti di entrambe le aziende si servono per il supporto alle decisioni, è pertanto fondamentale che questo tipo di inconvenienti non si verifichi o venga perlomeno ridotto al minimo. Per questa ragione, oltre alla creazione di un *data warehouse*, sono stati forniti al cliente strumenti tramite cui inserire i dati in modo da preservare la loro consistenza e limitare il più possibile lo sviluppo di errori. Vengono qui presentate le varie fasi di realizzazione del progetto: il caso di studio, il resoconto della riunione preliminare svolta con il committente, l'analisi e la specifica dei requisiti, la progettazione concettuale e logica dei *data mart*, l'analisi degli strumenti software utilizzati, la realizzazione delle procedure di estrazione, trasformazione e caricamento, la fase di progettazione e realizzazione della reportistica. Di ciascuna fase vengono introdotte e approfondite le problematiche di ordine generale, l'approccio seguito e i dettagli più importanti delle soluzioni adottate per risolverle.

INDICE

1	Introduzione	1
1.1	Presentazione del problema	1
1.2	Rassegna della letteratura	2
1.3	Contenuto della tesi	2
2	Il caso di studio	4
2.1	L'azienda committente: Accenture.....	4
2.1.1	Analisi e natura dei fini dell'azienda	5
2.2	L'azienda cliente: Velux.....	6
2.2.1	Analisi e natura dei fini dell'azienda	6
2.2.2	Accordi commerciali	7
2.3	Accenture Daylight Unit	7
2.4	Analisi dei processi aziendali	7
2.4.1	Registrazione delle ore di lavoro	8
2.4.2	Ticket di servizio	11
2.5	Struttura iniziale del sistema informativo	15
2.5.1	MSP	16
2.5.2	HP Service Center	16
2.5.3	Reportistica	17
2.5.4	Requisiti e richieste di nuove soluzioni	17
2.5.5	Possibile nuova struttura	19
3	Specifica dei requisiti di analisi e progettazione iniziale dei <i>data mart</i>	21
3.1	Data Warehousing	21
3.2	Registrazione delle ore	24
3.2.1	Specifica dei requisiti del fatto	24
3.2.2	Progettazione concettuale iniziale del <i>data mart</i>	34
3.3	Ticket di servizio	34
3.3.1	Specifica dei requisiti del fatto	34
3.3.2	Progettazione concettuale iniziale del <i>data mart</i>	44
3.4	Tabelle riepilogative delle dimensioni e delle misure	45

4	Progettazione concettuale finale e logica dei data mart e del <i>data warehouse</i>	46
4.1	Introduzione ai sistemi sorgenti	46
4.2	Presentazione della base di dati operativa MSP	48
4.3	MSP, classificazione delle tabelle interessanti	50
4.4	Accenture SharePoint e Master Data Services	51
4.5	Presentazione della base di dati operativa HPSC	52
4.6	HPSC, classificazione delle tabelle interessanti	52
4.7	KPI Targets	53
4.7.1	Progettazione concettuale iniziale del <i>data mart</i>	57
4.8	Progettazione logica del <i>data mart</i> Registrazione delle ore	58
4.9	Progettazione logica del <i>data mart</i> Ticket di Servizio	59
4.10	Progettazione logica del <i>data mart</i> KPI Targets	60
4.11	Progettazione logica del <i>data warehouse</i>	61
5	Ambiente e strumenti di sviluppo	62
5.1	Considerazioni sulla scelta dell'ambiente Microsoft	62
5.2	SQL Server 2012 Enterprise	63
5.2.1	Management Studio	64
5.2.2	Integration Services	64
5.2.3	Analysis Services	64
5.2.3.1	Modellazione Multidimensionale (SSAS)	65
5.2.3.2	Modellazione Tabulare (SSAS)	66
5.2.4	Reporting Services	68
5.2.5	Master Data Services	69
6	Procedure di estrazione, trasformazione e caricamento	74
6.1	Il processo di estrazione, trasformazione e caricamento dati	74
6.2	Vantaggi dell'utilizzo di SQL Server Integration Services (SSIS)	75
6.3	Struttura del pacchetto SSIS	77
6.3.1	Flusso di controllo	77
6.3.2	Flusso di dati	80
6.4	Nomenclatura	83
6.5	Staging Area	84
6.6	Fase di estrazione	85
6.7	Trasformazione e caricamento	86
6.8	Slowly Changing Dimensions	90
6.9	Creazione delle tabelle dei fatti	93
6.10	Esecuzione procedure e frequenza di aggiornamento	95
6.11	Schemi riassuntivi del flusso di dati	96

7	Esempi di reportistica	98
7.1	Reporting	98
7.2	SQL Server Reporting Services (SSRS)	99
7.3	Vantaggi dell'utilizzo di SQL Server Reporting Services (SSRS)	100
7.4	I report iniziali	101
7.5	I report realizzati	103
8	Conclusioni	107
	Bibliografia	109

INTRODUZIONE

1.1 Presentazione del problema

Accenture, multinazionale di consulenza direzionale, tramite il proprio reparto situato a Kolding, Danimarca, fornisce una vasta gamma di servizi di *Information Technology* a Velux, un'azienda danese specializzata nella produzione di pannelli e finestre. Il sistema informativo sul quale si fonda la reportistica prevede la presenza di due sistemi sorgenti, dai quali vengono estratti i dati che costituiscono la base delle analisi sull'efficienza del lavoro effettuato dai dipendenti di Accenture e sulle tematiche relative ai servizi offerti a Velux. Gran parte della reportistica, il cui fulcro è costituito da un certo numero di indicatori di performance definiti da Accenture, è realizzata manualmente da 9 FTE (*Full-time equivalent*, un metodo che viene usato frequentemente in ambito aziendale, di cui Accenture si serve per identificare il proprio personale) operanti in diverse postazioni sparse tra l'India e le Filippine e coordinati dal responsabile di progetto operante nella sede danese. L'inserimento manuale di una grossa mole di dati, congiuntamente alle problematiche legate alla difficoltà di coordinamento tra le diverse postazioni, rappresenta il fattore chiave alla base dello sviluppo di anomalie e incongruenze tra i dati. Partendo dall'analisi della soluzione attuale, l'obiettivo concordato con il committente è quello di procedere all'automazione dei processi di estrazione, trasformazione, gestione dei dati e di reportistica tramite la creazione di un *data warehouse*.

1.2 Rassegna della letteratura

Per la progettazione del *data warehouse*, in particolare l'approccio al problema, le tecniche di modellazione, la modulistica per la specifica dei requisiti e il procedimento di progettazione sono state consultate le soluzioni proposte da [Albano 12]. Per le caratteristiche dei software e degli ambienti utilizzati per lo sviluppo del progetto, per le procedure di estrazione, trasformazione e caricamento viene fatto ampio riferimento alla documentazione ufficiale Microsoft online contenuta nel sito [Msdn 13] e alle considerazioni proposte da Ruggieri nelle sue dispense [Ruggieri 12]. Per quanto riguarda la descrizione dei processi e l'introduzione delle aziende coinvolte viene fatto riferimento alla documentazione online presente nei siti [Accenture 13] e [Velux 13] e alla documentazione interna fornita da Accenture.

1.3 Contenuto della tesi

Il *data warehouse* è stato realizzato all'interno dell'azienda danese d60 e ha visto coinvolto un team di lavoro formato da (a) un Project Manager, figura responsabile dell'organizzazione e gestione del team di lavoro, con il compito di coordinare le varie fasi del progetto e assicurare alla committenza la corretta realizzazione degli obiettivi preposti entro i termini prestabiliti e (b) due consulenti di Business Intelligence, responsabili delle fasi di analisi, della specifica dei requisiti, della modellazione concettuale e logica del *data warehouse*, delle attività di progettazione e implementazione delle procedure di estrazione, trasformazione e caricamento, della creazione della reportistica e della manutenzione del sistema creato.

Il lavoro è stato lungo e impegnativo a causa di diversi fattori quali le difficoltà incontrate in merito alle rigide regole riguardanti la sicurezza e l'accessibilità dei dati vigenti in Accenture, le alte aspettative ed esigenze in termini di qualità ed efficienza da parte del committente e la dislocazione logistica delle postazioni coinvolte nel processo informativo. Parte del prodotto creato è orientato all'utilizzo da parte dei dipartimenti con sede in Asia, fattore il quale ha comportato numerosi incontri con il committente e comunicazioni con le diverse sedi.

L'obiettivo globale della tesi è quello di descrivere le fasi di progettazione e implementazione di un *data warehouse* basato sui dati riguardanti l'efficienza del lavoro effettuato dai dipendenti di Accenture e sulle tematiche relative ai servizi offerti a Velux. Esso rappresenta il fulcro sul quale si fonda la reportistica che permetterà ai responsabili di entrambe le aziende di condurre le corrette analisi per il supporto decisionale.

Nel dettaglio, il presente lavoro di tesi è organizzato nel modo seguente.

Il Capitolo 2 è dedicato alla descrizione del caso di studio, vengono descritti brevemente i contesti aziendali di riferimento e viene introdotto il lettore al progetto, presentando la struttura del precedente sistema informativo e gli obiettivi di partenza prefissati durante l'incontro preliminare svolto con il committente.

Il Capitolo 3 è dedicato alla progettazione iniziale del *data warehouse*, partendo dalla specifica dei requisiti, fino ad arrivare al disegno concettuale iniziale dei *data mart* interessanti. Per fare questo vengono identificati i fatti, le dimensioni e le misure.

Il Capitolo 4 è dedicato alla progettazione finale del *data warehouse*, partendo dall'analisi della base dati operativa, realizzando gli schemi concettuali e logici finali dei *data mart* e infine progettando lo schema logico del *data warehouse*.

Il Capitolo 5 è dedicato alla presentazione degli strumenti utilizzati per lo sviluppo del progetto. Vengono spiegati i motivi che hanno spinto all'utilizzo della piattaforma Microsoft per la *Business Intelligence* e ne vengono descritte le caratteristiche principali, con particolare attenzione a quelle maggiormente utilizzate per lo sviluppo del *data warehouse*.

Nel Capitolo 6 vengono introdotti i processi di estrazione, trasformazione e caricamento. Si descrivono i punti salienti della realizzazione, focalizzando l'attenzione sulla descrizione dei problemi riscontrati nel corso dell'implementazione di tali procedure e le relative soluzioni adottate.

Nel Capitolo 7, infine, vengono presentati alcuni tra i report realizzati.

IL CASO DI STUDIO

Questo capitolo è dedicato alla presentazione delle aziende coinvolte nel processo di creazione del *data warehouse*, alla terminologia, alle logiche e alle analisi sull'efficienza del lavoro effettuato dai dipendenti di Accenture e sulle tematiche relative ai servizi offerti a Velux. Durante la descrizione dei processi viene presentato il sistema precedentemente in uso tramite un approfondito resoconto della riunione preliminare svolta con il committente e vengono definiti gli indicatori di performance atti a permettere a ciascun responsabile di trarre le corrette valutazioni sull'operato del personale.

Lo scopo del capitolo è quello di introdurre i punti salienti del progetto per motivare le analisi dei requisiti che, nel capitolo successivo, vengono documentate in modo dettagliato.

2.1 L'azienda committente: Accenture

Accenture è un'azienda globale di consulenza direzionale, servizi tecnologici e outsourcing che conta oltre 260.000 professionisti in più di 120 paesi nel mondo. Vantando una lunga esperienza e competenze in tutti i settori di mercato, Accenture collabora con alcune tra le più importanti aziende e pubbliche amministrazioni del mondo e con organizzazioni di qualsiasi dimensione e settore industriale, con lo scopo di raggiungere performance d'eccellenza. In particolare, nell'anno 2012, annovera tra i suoi clienti 89 delle 100 aziende della classifica Fortune Global 100 e oltre tre quarti delle Fortune Global 500. Accenture punta a ottimizzare la soddisfazione del cliente come indice di garanzia per il consolidamento e l'estensione delle proprie collaborazioni.

Come ribadito in [Accenture 2013], tra i punti cardine della politica aziendale è doveroso dare risalto all'eccellenza del servizio tramite la quale si punta, grazie a un solido lavoro di squadra e una

rete estesa su scala mondiale, a migliorare le performance di business dei clienti e si cerca di instaurare con loro relazioni durature.

2.1.1 Analisi e natura dei fini dell'azienda

La strategia di Accenture per un business ad alte performance, ampiamente descritta da [Accenture 13], mette a frutto le competenze maturate dall'azienda nel business consulting, nell'innovazione tecnologica e nei servizi in outsourcing per aiutare i clienti a raggiungere performance di eccellenza, generando valore sostenibile per le aziende e i loro azionisti. Sfruttando la profonda conoscenza del settore, l'esperienza nell'offerta di servizi e un vasto patrimonio di capacità tecnologiche avanzate, Accenture punta a identificare le nuove tendenze di business e le tecnologie innovative per sviluppare soluzioni in grado di supportare i clienti di ogni parte del mondo a penetrare nuovi mercati, aumentare il profitto nei mercati in cui già operano, migliorare le performance operative e fornire i loro prodotti e servizi in modo più efficiente ed efficace.

Tramite lo sviluppo di una solida rete di alleanze, Accenture mira a supportare i clienti nell'innovazione di tutti gli aspetti del loro business, dalla pianificazione strategica alle attività quotidiane, con l'obiettivo centrale di aiutarli a diventare aziende ad alte performance e ottenere al tempo stesso il massimo valore dagli investimenti nella tecnologia.

La rete di alleanze rafforza la capacità dell'azienda di aiutare i clienti a ottenere la riduzione sostenibile dei costi IT, la riduzione del rischio associato all'erogazione dei servizi, la rapida identificazione dei fornitori, con risorse di assistenza dedicate, l'accesso privilegiato a software di sviluppo, hardware dimostrativo, esperienza a livello di architettura, assistenza nel dimensionamento e nella configurazione.

2.2 L'azienda cliente: Velux

Velux è un'azienda danese specializzata nella produzione di pannelli e finestre, ad oggi uno dei marchi più conosciuti nel settore dei materiali per l'edilizia. Secondo le statistiche riportate da [Velux 13], è presente in 11 paesi con stabilimenti produttivi e globalmente in 40 paesi, fra cui l'Italia, con società di vendita in cui operano complessivamente circa 10.000 dipendenti. Velux venne fondata nel 1941 quando Villum Kann Rasmussen intravide per la prima volta delle ottime opportunità commerciali nello sviluppo di sottotetti a basso costo. Rasmussen scelse il nome Velux in quanto, oltre a essere distintivo e facile da pronunciare e memorizzare, è una combinazione delle parole “Ve”, abbreviazione del termine inglese “Ventilation” e “Lux”, dal latino, luce. Nel 1942 Velux lanciò nel mercato la prima finestra per tetto con telaio rivestito in metallo. A partire da quel prototipo, l'azienda ha continuato a innovare e migliorare le prestazioni e sostenibilità dei propri prodotti fino a espandere il proprio mercato a livello mondiale.

2.2.1 Analisi e natura dei fini dell'azienda

Negli ultimi settanta anni Velux ha sviluppato esperienza in ambito energetico e architettonico. “Partecipa attivamente a progetti internazionali per la realizzazione di edifici sostenibili e a zero impatto ambientale, caratterizzati da un comfort abitativo ottimale” [Velux 13]. Velux mira a migliorare le condizioni abitative e lavorative puntando su luce naturale e aria fresca, elementi che fanno parte del vivere quotidiano in milioni di case di tutto il mondo.

La gamma di prodotti comprende un'ampia varietà di finestre per tetti, tende, persiane, comandi elettrici, componenti originali per finestre da tetto. L'azienda propone inoltre finestre cupolino per tetti piani, tunnel solari e collettori solari termici.

2.2.2 Accordi commerciali

Accenture e Velux hanno siglato un accordo commerciale nel 2005 in base al quale Velux si impegna ad affidare interamente le proprie attività di Information Technology ad Accenture. La maggior parte del personale operante all'epoca nella sezione IT è stato trasferito in Accenture, sebbene un gruppo più ridotto di impiegati sia rimasto in Velux in modo tale da mantenere operative le postazioni di lavoro preesistenti.

2.3 Accenture Daylight Unit

La Accenture Daylight Unit (ADU), con sede operativa a Kolding, Danimarca, opera su una estesa piattaforma internazionale le cui risorse sonolocate sia a livello locale che in svariate postazioni sparse tra l'Europa e l'Asia (India, Spagna e Filippine sono le principali). È l'unità maggiormente coinvolta nei processi legati alla realizzazione del *data warehouse*, in quanto si occupa della gestione delle informazioni relative alla collaborazione con Velux.

In particolare, le sedi di Bangalore, Hyderabad e Manila hanno diretto accesso e curano una vasta mole di dati, a partire dai quali generano manualmente a cadenza settimanale un numero prestabilito di report.

2.4 Analisi dei processi aziendali

Nel corso della riunione preliminare con il committente, oltre agli aspetti tecnici che verranno approfonditi in seguito, sono stati introdotti i concetti chiave legati ai due processi considerati strategici per l'attività operativa del reparto ADU, tanto da necessitare di uno strumento analitico per il supporto alle decisioni.

Per ogni processo interessato, vengono introdotti i termini rilevanti nella terminologia aziendale e vengono identificati e raccolti, in modo informale, i requisiti di analisi dei dati, con particolare attenzione agli indicatori di performance, che i responsabili di funzione ed i manager di alto livello sono interessati a consultare per supportare le loro decisioni.

2.4.1 Registrazione delle ore di lavoro

Il primo dei due processi coinvolti nel progetto riguarda la registrazione delle ore di lavoro del personale Accenture appartenente per lo più al reparto ADU. A cadenza settimanale, vengono monitorati il tempo e le ore di lavoro dei dipendenti assegnati ai vari progetti attivi al fine di valutarne l'operato e la produttività in base a degli indicatori di performance e al loro confronto con un determinato insieme di valori “*target*” di riferimento.

Accenture ha stabilito delle regole interne per la registrazione delle ore di lavoro, una suddivisione gerarchica del personale, degli standard per quanto riguarda i carichi settimanali di lavoro e una logica organizzativa dei progetti attivi.

Per quanto riguarda l'organizzazione gerarchica, il gradino più elevato viene definito “*Organization*”, del quale fanno parte il reparto ADU di Accenture e una generica categoria “Non Accenture” utilizzata per indicare sia il personale Velux che eventuali collaboratori esterni coinvolti nei processi. Il livello successivo è definito “*Tower*”, mentre gli ultimi due livelli della gerarchia sono i “*Team*” e i “*Subteam*”, dei quali fanno parte le singole risorse umane.

Un'altra importante distinzione è di tipo logistico e si riferisce alle postazioni in cui opera il personale. Anche in questo caso viene fatta distinzione tra due categorie, *On-Shore* o *On-Site* (postazioni danesi) e *Off-Shore* o *IDC* (postazioni sparse nel resto del mondo).

Accenture utilizza l'acronimo WBS, che in generale significa *Work Breakdown Structure*, indica l'insieme di attività di cui si compone un progetto ed è un termine frequentemente utilizzato nel Project Management per coadiuvare il Project Manager nell'organizzazione delle attività di cui è responsabile, per catalogare le proprie attività. Tra i valori che assume è doveroso citare “Project”, in riferimento ai generici progetti, “Internal Time”, in riferimento al tempo impiegato per risolvere problematiche interne all'azienda e “Unassigned Time”, in riferimento alle ore di lavoro non assegnate a nessuna delle precedenti categorie, essendo questi i valori interessanti ai fini del calcolo degli indicatori di performance.

I compiti assegnati ai dipendenti possono essere racchiusi all'interno di un progetto o possono essere dei semplici task, che, in generale, sono più specifici rispetto ai progetti. Sebbene si possa pensare a un'associazione tra questi due tipi di funzioni, Accenture le considera attività separate. Non esistono regole a priori sulla nomenclatura dei task, delegando ai singoli Subteam il compito di assegnare ai task un nome consono al contesto nel quale si opera. Questo aspetto verrà approfondito in seguito, dal momento che la creazione di un algoritmo associativo è stata parte integrante del progetto. Infine è doveroso specificare ai fini delle analisi sull'altro processo coinvolto nella

creazione del *data warehouse* che alcuni task sono dedicati alla risoluzione di particolari ticket i cui dettagli verranno elencati in seguito.

Per quanto concerne, infine, le regole interne di registrazione delle ore, ogni dipendente è tenuto a compilare un foglio Excel a cadenza settimanale, con scadenza alle 23:59 della domenica. I dati accumulati entro quel termine vengono utilizzati per la stesura del report settimanale.

Raccolta dei requisiti della Registrazione delle ore di lavoro

Il reparto ADU divide i propri indicatori di performance in due gruppi. Vengono ora elencati, in termini di requisiti di analisi, quelli relativi a questo processo, appartenenti al primo gruppo denominato *Sustainable Profitability*. Col il termine FTE, acronimo introdotto nel primo capitolo di questo documento, in questo contesto si intende un'unità di misura definita come le ore di lavoro registrate in relazione alle ore lavorative previste (nella terminologia della documentazione Accenture si parla di "*Actual Hours*"). I manager hanno interesse ad analizzare i valori degli indicatori partendo dal livello gerarchico Subteam fino a raggiungere i livelli più alti, pertanto, sebbene non venga specificato nella tabella sottostante, ogni requisito di analisi fa riferimento a uno specifico Subteam, Team, Tower o Organization, a seconda del tipo di report prodotto.

N.	Nome	Requisito di analisi
1	Base Service On-Site Effort	Numero di FTE associati a progetti il cui WBS Type corrisponde a <i>Base Service</i> , svolti da risorse operanti <i>On-Site</i> , per un dato livello della gerarchia organizzativa e per data.
2	Base Service IDC Effort	Numero di FTE associati a progetti il cui WBS Type corrisponde a <i>Base Service</i> , svolti da risorse operanti <i>Off-Shore</i> , per un dato livello della gerarchia organizzativa e per data.
3	ME On-Site Effort	Numero di FTE associati a progetti il cui WBS Type corrisponde a <i>Minor Enhancements</i> , svolti da risorse operanti

		<i>On-Site</i> , per un dato livello della gerarchia organizzativa e per data.
4	ME IDC Effort	Numero di FTE associati a progetti il cui WBS Type corrisponde a <i>Minor Enhancements</i> , svolti da risorse operanti <i>Off-Shore</i> , per un dato livello della gerarchia organizzativa e per data.
5	Average Hourly Rate per ME	Costo medio orario relativo a progetti il cui WBS Type corrisponde a <i>Minor Enhancements</i> , per un dato livello della gerarchia organizzativa e per data.
6	Unassigned Time	Numero di FTE associati a task di tipo <i>Unassigned time</i> , per un dato livello della gerarchia organizzativa e per data.
7	Internal Time	Numero di FTE associati a progetti il cui WBS Type corrisponde a <i>Internal time</i> , per un dato livello della gerarchia organizzativa e per data.
8	Project Effort On-Shore	Numero di FTE associati a progetti svolti da risorse operanti <i>On-Site</i> , per un dato livello della gerarchia organizzativa e per data.
9	Project Effort Off-Shore	Numero di FTE associati a progetti svolti da risorse operanti <i>Off-Shore</i> , per un dato livello della gerarchia organizzativa e per data.
10	IDC Contribution	Percentuale di lavoro associata a progetti il cui WBS Type corrisponde a <i>Minor Enhancements</i> svolti da risorse operanti <i>Off-Shore</i> , per un dato livello della gerarchia organizzativa e per data.
11	Actual Work Spend On BaseService And FixOnFail	Numero di ore lavorative fatturabili associate a progetti il cui WBS Type corrisponde a <i>Base Service</i> e task di tipo <i>Unassigned time</i> , per un dato livello della gerarchia organizzativa e per data.

2.4.2 Ticket di servizio

Il secondo processo coinvolto riguarda la creazione, gestione e chiusura dei cosiddetti “Ticket” di servizio, utilizzati da Accenture per gestire e monitorare problematiche, anomalie o eventuali richieste legati all’attività del proprio personale e del personale Velux. Vengono divisi in categorie e in particolare Accenture definisce tre tipologie di ticket a seconda del contesto al quale si riferiscono:

- **Requests.** Marcati con il prefisso Q, sono gli unici ticket che fanno riferimento ad attività legate ai task di cui discusso in 2.4.1.
- **Incidents.** Marcati con il prefisso IM, fanno riferimento ad anomalie sorte in relazione ad altre attività (interne e non) svolte dal personale Accenture/Velux.
- **Problems.** Marcati con il prefisso PM, fanno riferimento a problematiche sorte in relazione ad altre attività (interne e non) svolte dal personale Accenture/Velux.

Ogni ticket attraversa diverse fasi del proprio ciclo di vita. Sono 19 quelle previste dalla documentazione ufficiale Accenture, sebbene in fase di reporting vengano racchiuse in 3 maggiori categorie:

- **Open.** È lo status in cui si trova il ticket quando viene creato e permane in questo stato finché non viene risolto. Il ticket viene considerato “open” anche quando modificato (updated) o in stato di allerta (alert);
- **Resolved.** È lo status che precede la chiusura. Il problema/incidente per cui il ticket era stato creato è stato risolto o la richiesta è stata esaudita;
- **Closed.** Lo stato finale, indica che il ticket è definitivamente chiuso.

I ticket vengono assegnati, gestiti e chiusi da dipendenti di Accenture o Velux a seconda del contesto nel quale si verifica l’anomalia. Le logiche di assegnamento dei ticket, tuttavia, non sono state rese note dal committente in quanto non incidono sulla realizzazione del *data warehouse*.

Raccolta dei requisiti del Ticket di servizio

Gli indicatori di performance correlati al processo Ticket di servizio sono quelli appartenenti al secondo gruppo, detto “*Operational Excellence*”. Il loro scopo è quello di monitorare il ciclo di vita dei ticket e accertare che le varie anomalie, richieste e problematiche vengano risolte entro determinati limiti temporali in modo tale da assicurare il mantenimento di un certo standard di servizio. Per quanto riguarda i soggetti di interesse vale lo stesso concetto enunciato in relazione al processo Registrazione delle ore di lavoro: i manager hanno interesse ad analizzare i valori degli indicatori partendo dal livello gerarchico Subteam fino a raggiungere i livelli più alti, pertanto, sebbene non venga specificato nella tabella sottostante, ogni requisito di analisi fa riferimento a uno specifico Subteam, Team, Tower o Organization, a seconda del tipo di report prodotto.

N.	Nome	Requisito di analisi
12	Incidents Resolved in Due Time	Percentuale di ticket di tipo Incident risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.
13	Overdue Incidents	Numero di ticket di tipo Incident non risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.
14	Average Age (Incidents)	Numero medio di giorni di vita dei ticket di tipo Incident che non sono stati risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.
15	ME's Resolved in Due Time	Percentuale di ticket di tipo Request risolti entro i termini prestabiliti in riferimento, per un dato livello della gerarchia organizzativa e per data.
16	Overdue ME's	Numero di ticket di tipo Request non risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.
17	Average Age (ME)	Numero medio di giorni di vita dei ticket di tipo Request che non sono stati risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.

18	Average Time Spend (ME)	Numero medio di ore spese per risolvere un ticket di tipo Request, per un dato livello della gerarchia organizzativa e per data.
19	ME Backlog	Numero di ore preventivate per la risoluzione dei ticket di tipo Request, per un dato livello della gerarchia organizzativa e per data.
20	Budget Overruns (%)	Percentuale di ticket di tipo Request che sono stati risolti in un numero di ore maggiore rispetto a quelle preventivate, per un dato livello della gerarchia organizzativa e per data.
21	Average Budget Overrun	Numero medio di ore in eccesso dei ticket di tipo Request che sono stati risolti in un numero di ore maggiore rispetto a quelle preventivate, per un dato livello della gerarchia organizzativa e per data.
22	Days Since Last Update (ME)	Numero di ticket di tipo Request la cui data relativa all'ultimo aggiornamento eccede i 30 giorni, per un dato livello della gerarchia organizzativa e per data.
23	Days Since Last Update (Incidents)	Numero di ticket di tipo Incident la cui data relativa all'ultimo aggiornamento eccede i 7 giorni, per un dato livello della gerarchia organizzativa e per data.
24	Open Problems	Numero di ticket di tipo Problem marcati come Open, per un dato livello della gerarchia organizzativa e per data.
25	Overdue Problems	Numero di ticket di tipo Problem non risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.
26	VIP Incidents in Due Time	Percentuale di ticket di tipo Incident associati a una particolare categoria di persone definita Critical User risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.
27	VIP Requests in Due Time	Percentuale di ticket di tipo Request associati a una particolare categoria di persone definita Critical

		User risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.
28	Broken Lead Time for Estimation	Numero assoluto di ticket di tipo Request che sono stati risolti in un numero di ore maggiore rispetto a quelle preventivate, per un dato livello della gerarchia organizzativa e per data.

2.5 Struttura iniziale del sistema informativo

La riunione preliminare, svolta in presenza del responsabile del reparto ADU per la Danimarca, è stata fondamentale per comprendere le logiche del sistema informativo alla base dei processi descritti in precedenza e le richieste del committente riguardo alla creazione del *data warehouse*. Sono stati evidenziati gli aspetti tecnici legati alle sorgenti dei dati, gli aspetti logistici legati alle postazioni delegate alla creazione e gestione dei report, le dinamiche dei processi, le tecnologie utilizzate e le esigenze del committente riguardanti gli aspetti tecnici e le tempistiche del progetto. La Figura 2.1 mostra la struttura base del sistema informativo iniziale.

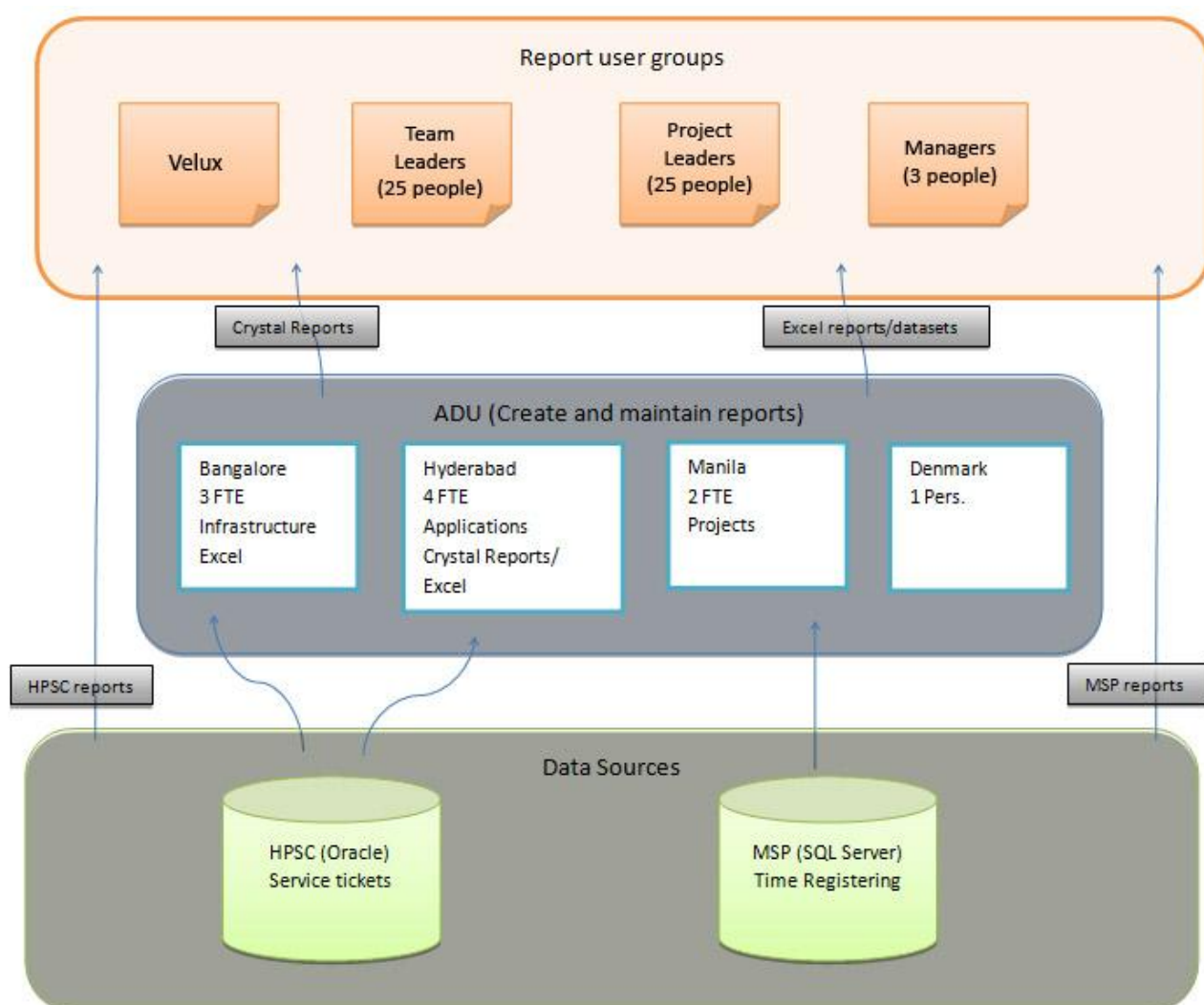


Figura 2.1: Struttura del sistema informativo iniziale

Come si evince dalla Figura, i dati estratti dai due sistemi sorgente vengono utilizzati dalle varie postazioni appartenenti al reparto Accenture Daylight Unit per creare e gestire i report per mezzo degli strumenti Excel e Crystal Reports. I report generati vengono consultati da managers, team leaders, project leaders e responsabili di Velux.

2.5.1 MSP

Si tratta di un Microsoft SQL Server che contiene, tra le altre cose, i dati relativi al monitoraggio del tempo e delle ore di lavoro dei dipendenti. Dalla riunione preliminare non sono emerse informazioni sul volume di dati contenuti nel server MSP. Il sistema è stato realizzato nel mese di ottobre 2011 e non sono presenti dati storici antecedenti questa data.

Per quanto riguarda la frequenza di aggiornamento, una volta raccolti i dati inviati dal personale, il server viene aggiornato due volte alla settimana: la domenica allo scadere del termine massimo indicato in precedenza e il martedì. I report della settimana conclusa vengono creati il lunedì, pertanto i dati raccolti il martedì incidono esclusivamente sui report della settimana successiva. L'idea base, per quanto riguarda il *data warehouse*, è quella di eseguire le procedure di aggiornamento delle tabelle dimensionali e delle tabelle dei fatti due volte alla settimana, prelevando i dati dal server dopo che quest'ultimo è stato aggiornato.

2.5.2 HP Service Center

HPSC (HP Service Center) è un server Oracle in cui vengono registrati i dati relativi ai progetti e ai cosiddetti "Service Tickets", divisi in categorie (Incidents, Problems, Requests). I dati presenti nel server vengono aggiornati continuamente e la loro quantità cresce dell'ordine di 80.000 righe all'anno. Il server ha iniziato a immagazzinare dati dal 2009. L'obiettivo concordato in sede di riunione preliminare è quello di aggiornare il futuro *data warehouse* quotidianamente. Non è stata tuttavia fatta chiarezza su quali procedure adottare per l'estrazione dei dati dalle tabelle operazionali e questo aspetto ha inciso pesantemente sulle tempistiche di realizzazione del *data warehouse* e ha complicato le fasi di analisi antecedenti la progettazione vera e propria.

2.5.3 Reportistica

La reportistica attuale viene gestita principalmente dal reparto ADU, il quale genera più di 200 report (la maggior parte dei quali eseguiti automaticamente o in modo semi-automatico e realizzati estraendo dati dalle sorgenti di origine) servendosi dello strumento “Crystal Reports”. Vi sono anche report manuali, che risultano essere pesanti e richiedono una grande quantità di lavoro (mapping, filtering, definizione di business rules), orientati soprattutto al monitoraggio e all’analisi interna. Uno dei principali report manuali è costituito dall’analisi dei KPI. Il cliente, Velux, riceve report esclusivamente relativi ai dati presenti nel server HPSC, dal momento che quelli contenuti nel server MSP, essendo di uso interno, sono rilevanti solo per i responsabili Accenture.

Oltre ai report standard, il reparto ADU produce anche una grande quantità di report ad hoc destinati ai Team Leader, i quali hanno esigenze temporali di consultazione entro determinate scadenze. Come discusso in precedenza, i dati vengono estratti direttamente dalle fonti di origine e la mole di lavoro manuale è una potenziale fonte di errore non trascurabile. Oltretutto la generazione dei report è affidata a nove dipendenti, operanti in tre postazioni diverse, fattore il quale, date le difficoltà di coordinamento, favorisce lo sviluppo di incongruenze tra i dati.

2.5.4 Requisiti e richieste di nuove soluzioni

Data warehouse

L'automazione è il requisito fondamentale richiesto dal committente. L'estrazione dei dati dai sistemi sorgente dovrà essere quanto più possibile automatizzata. Uno degli obiettivi fondamentali è che la gestione/modifica dei dati e il mapping risulti essere facile e accessibile. Un altro aspetto chiave riguarda la gestione dei cambiamenti storici, fino a quel momento trascurata, la quale, in base a quanto concordato, dovrà essere implementata nelle aree richieste.

Reportistica

Tutti i report generati in precedenza con Crystal Reports e con Excel devono essere modificati (nuova origine dati) o ricreati nei nuovi strumenti di reportistica. In particolare, la nuova soluzione deve supportare i seguenti tipi di report:

- Report permanenti;
- Report Ad hoc / pivot per le analisi dinamiche;
- Self-service Business Intelligence;

- Analisi Key Performance Indicator.

In particolare si prospetta di focalizzarsi su:

- Maggiore accessibilità alle analisi ad hoc per gli utenti finali;
- Definizioni di figure chiave e della IPC (intern-process communication);
- Automazione, garanzia della qualità, tracciabilità e accuratezza della documentazione.

La sicurezza non rappresenta un punto chiave, dal momento che la politica di Accenture in questo progetto è che i dati in questione siano, in linea generale, a disposizione di tutte le persone coinvolte. Velux, cliente esterno con accesso ad alcuni dati del server HPSC, rappresenta un'eccezione, e in ogni caso, come ribadito, non ha accesso ai dati contenuti nel server MSP.

2.5.5 Possibile nuova struttura

La Figura 2.2 mostra, in base alle premesse fatte, la possibile nuova struttura del sistema informativo una volta terminato il progetto.

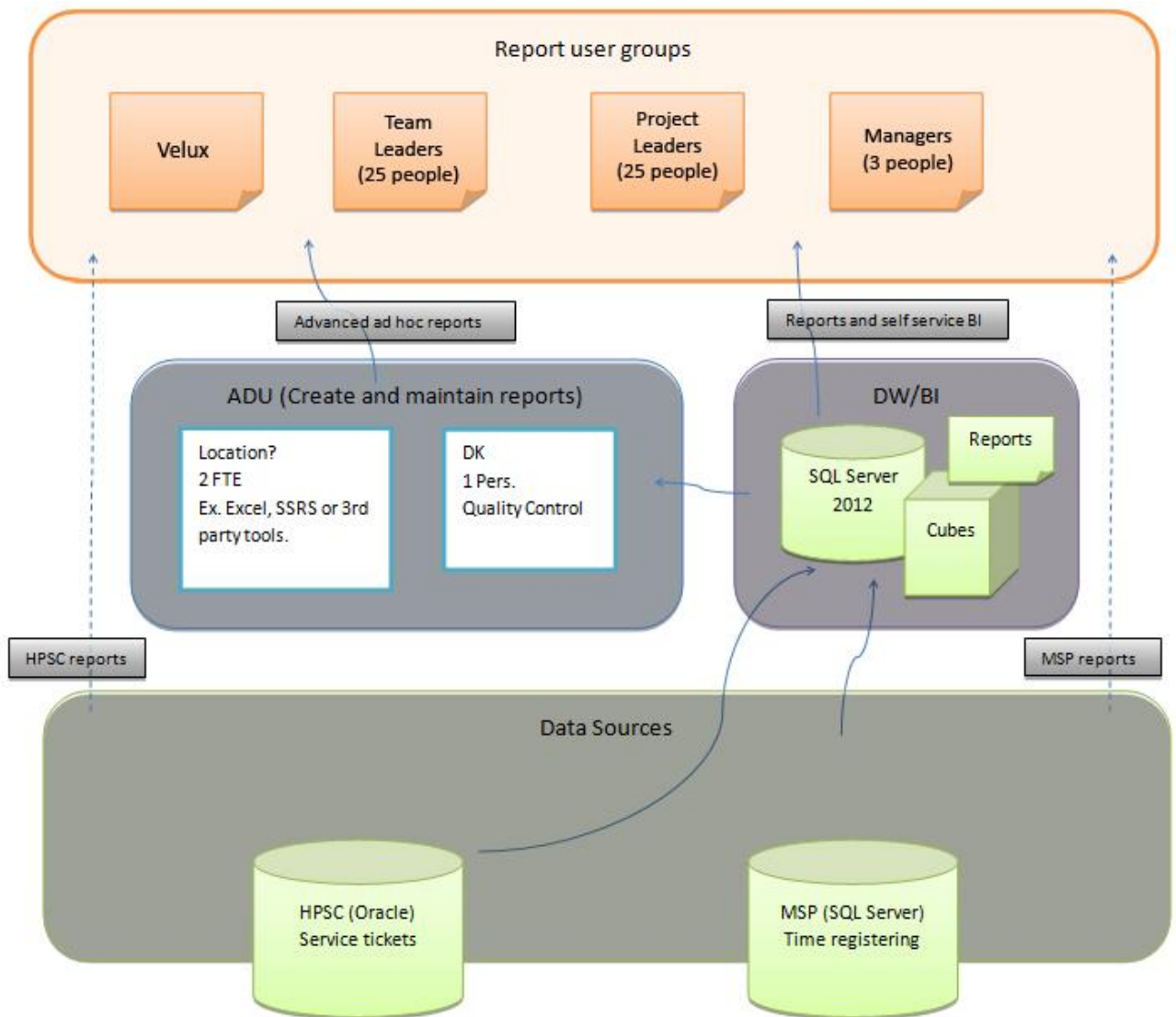


Figura 2.2: Possibile nuova struttura del sistema informativo

La Figura evidenzia che i dati raccolti dai sistemi sorgente vengono elaborati e utilizzati per popolare le tabelle dimensionali e le tabelle dei fatti, a partire dalle quali verranno generati, tramite specifici strumenti (in particolare SQL Server Reporting Services, del quale si parlerà in seguito), i report generici. Per quanto riguarda i report ad hoc, le varie postazioni del reparto ADU avranno la possibilità di crearli estraendo i dati direttamente dal *data warehouse* e non più dai sistemi sorgente.

Riassumendo, il cuore della soluzione è la costruzione di un *data warehouse*, con aggiornamento

automatico dei dati estratti dai server HPSC e MSP. Saranno curati con particolare attenzione gli aspetti riguardanti la mappatura dei dati, la creazione di business rules, le misure calcolate e la reportistica associata.

Partendo dalla base di un solido *data warehouse*, esiste la possibilità di utilizzare e fornire diversi strumenti di analisi e reporting. Verrà valutata l'eventualità di utilizzare lo strumento di reportistica integrato in Microsoft SQL Server (Reporting Services) e servirsi di Microsoft Office (Excel). Entrambi gli strumenti possono essere facilmente integrati da strumenti di terze parti come ad esempio "Targit" o "QlikView". La Figura 2.3 mostra un prospetto del potenziale flusso di dati.

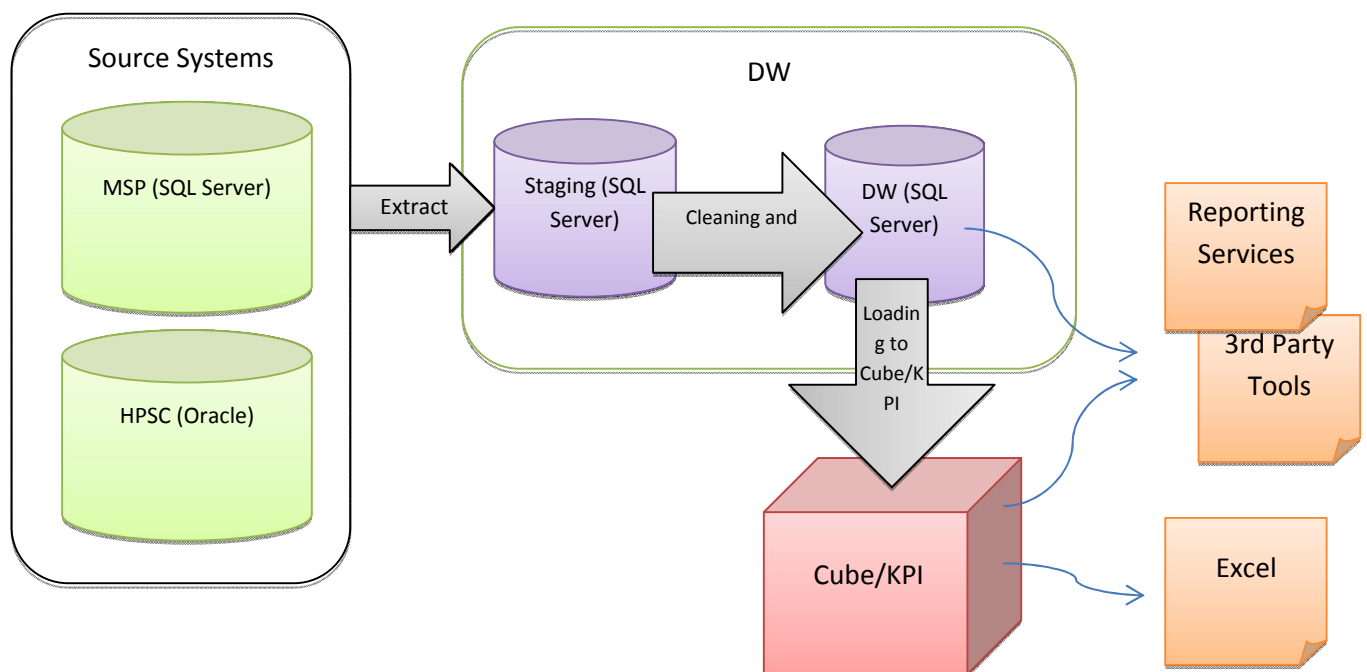


Figura 2.3: Prospetto del potenziale flusso di dati.

SPECIFICA DEI REQUISITI DI ANALISI E PROGETTAZIONE INIZIALE DEI *DATA MART*

Dopo una breve introduzione sulle scelte, caratteristiche e aspetti architetturali del processo di data warehousing, viene documentata formalmente la progettazione concettuale del *data warehouse* a partire dalle analisi dei requisiti. In particolare, l'attenzione viene focalizzata sulle specifiche dei requisiti e sul disegno concettuale iniziale dei *data mart* sulla base dei requisiti di analisi.

3.1 Data warehousing

Col il termine *data warehouse* si intende una particolare base di dati con le seguenti principali caratteristiche [Albano12]:

- **Tempificata.** Contiene informazioni sul tempo in cui si verificano gli eventi di interesse. Questa caratteristica permette di realizzare la storicizzazione delle informazioni.
- **Integrata.** I dati non provengono in genere da un'unica sorgente (una base di dati dell'organizzazione), ma sono il risultato di un lungo e costoso processo di integrazione di dati eterogenei.
- **Statica.** I dati vengono usati interattivamente per operazioni di ricerca e non di modifica. Periodicamente ai dati disponibili se ne aggiungono di nuovi o si rimuovono quelli ritenuti obsoleti.
- **Organizzata per soggetti.** Nei sistemi operazionali i dati sono organizzati per eseguire le attività aziendali quotidiane, mentre nei sistemi direzionali i dati sono organizzati per analizzare dei soggetti di interesse che influenzano l'andamento complessivo dell'azienda.

Quando i dati riguardano un solo soggetto di interesse si parla di *data mart* e possono essere un sottoinsieme di un *data warehouse* più generale.

- **Finalizzata ad analisi di supporto alle decisioni.** I dati sono organizzati per facilitare le loro analisi al fine di produrre delle opportune sintesi di supporto ai processi decisionali.

Il termine data warehousing si usa per riferirsi al processo utilizzato per organizzare i dati in un *data warehouse* e consentire poi agli utenti di analizzarli con strumenti di Business Intelligence.

Per la realizzazione del processo di data warehousing è stata adottata un'architettura multilivello che prevede tre livelli di memorizzazione dei dati: il livello delle sorgenti di dati, il livello di staging area che contiene i dati ottenuti dal processo di integrazione delle diverse sorgenti di dati e il livello del *data warehouse* [Albano12]. Nello specifico, la staging area integra tutte le informazioni provenienti dai diversi sistemi di archiviazione aziendali dopo che queste ultime, in seguito alla fase di estrazione, trasformazione e caricamento (ETL) sono state selezionate, ripulite e rese omogenee, al netto però di tutte le tabelle ed attributi giudicati non utili in fase di analisi dei processi e dei sistemi informativi. I dati della staging area possono essere aggiornati continuamente come quelli operazionali, e trasferiti poi quando preventivato nel *data warehouse*.

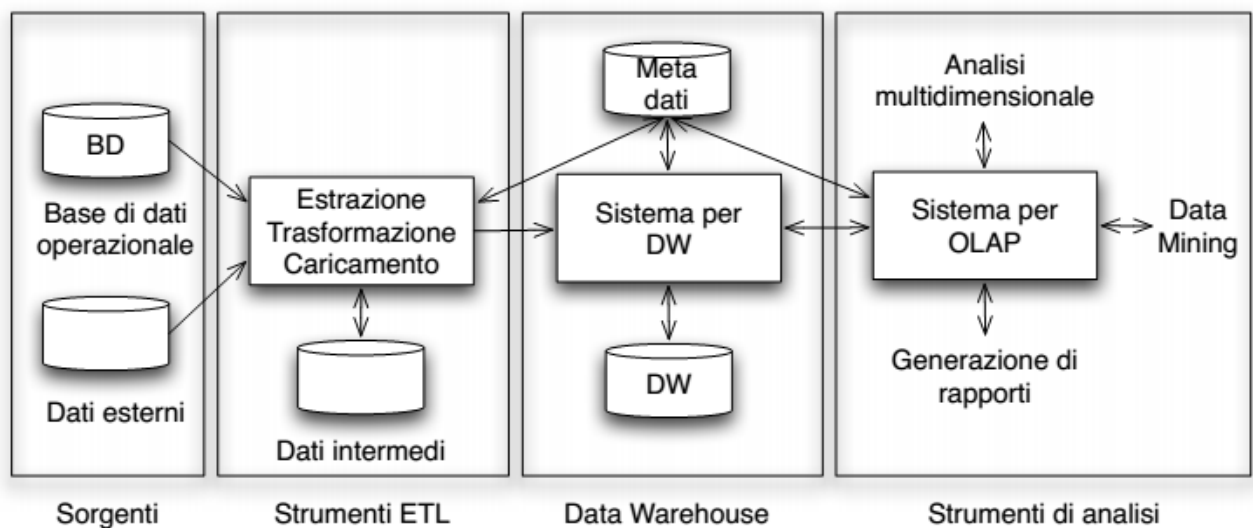


Figura 3.1: Architettura a tre livelli del processo di data warehousing [Albano 12].

Come mostrato in Figura 3.1, il processo di data warehousing è composto da quattro fasi principali:

1. **Sorgenti.** Definizione delle sorgenti dei dati, base di dati operativa ed eventualmente fonti esterne.
2. **Extraction Trasformation Loading.** Fase in cui i dati provenienti dalle sorgenti vengono opportunamente estratti, modificati e caricati nel sistema per *data warehouse*.
3. **Data warehouse.** Sistema in cui sono opportunamente memorizzati i dati con lo scopo di ottimizzare la loro successiva analisi.
4. **Strumenti di analisi.** Insieme di strumenti che interrogando il *data warehouse* permettono di creare reportistica ad-hoc, analizzare i dati in modo multidimensionale ed effettuare operazioni di data mining.

Una volta creato e popolato, il *data warehouse*, come discusso in precedenza, necessiterà di fasi di aggiornamento effettuate ad intervalli di tempo regolari per garantire l'allineamento con la base di dati operativa.

Per evitare di incorrere in inconvenienti nel corso della progettazione del *data warehouse*, come suggerito in [Albano 12], è stato stabilito di procedere, fin dall'incontro con il committente, considerando congiuntamente sia i requisiti di analisi che la base di dati operativa. Questo per scongiurare il rischio che:

- Venissero trascurati alcuni importanti requisiti di analisi;
- Venissero proposte al committente delle soluzioni potenzialmente non realizzabili per mancanza dei dati operativi necessari.

Pertanto le fasi seguite per la progettazione del *data warehouse* sono le seguenti:

1. Analisi dei requisiti.
2. Progettazione concettuale iniziale dei *data mart*.
3. Progettazione concettuale dei *data mart* dai dati operativi.
4. Progettazione concettuale finale dei *data mart*.
5. Progettazione logica dei *data mart* e del *data warehouse*.

All'interno di questo documento la raccolta dei requisiti viene inglobata all'interno delle analisi dei processi aziendali in modo da spiegare un passo per volta le scelte fatte in relazione ai processi stessi.

Nel presente capitolo, successivamente alla specifica dei requisiti, viene affrontata la progettazione concettuale iniziale dei *data mart* a partire dai requisiti di analisi, mentre nel capitolo successivo, dopo la presentazione della base dati operativa, viene trattata la progettazione concettuale finale dei *data mart* e la progettazione logica dei *data mart* e del *data warehouse*.

Prima di passare alla specifica dei requisiti e alle successive fasi di progettazione vengono definiti, partendo dai processi descritti in precedenza, i due fatti interessanti sui quali ruoterà la successiva progettazione, ovvero la registrazione delle ore di lavoro e il processo ticket di servizio. D'ora in avanti ciascun fatto viene trattato singolarmente, documentando in modo formale la progettazione concettuale e logica del relativo *data mart*. In conclusione viene definito lo schema logico complessivo del *data warehouse*.

3.2 Registrazione delle ore

Si descrive la specifica dei requisiti e la progettazione concettuale iniziale del *data mart* *Registrazione delle ore*. Con “registrazione delle ore” si intende la registrazione delle ore di lavoro svolte in una giornata da un dipendente appartenente a un Subteam in relazione a uno specifico task o a un progetto riguardante una determinata categoria.

3.2.1 Specifica dei requisiti del fatto

Nella fase di specifica dei requisiti sono stati prodotti documenti formali dei requisiti di analisi dei dati richiesti che ne evidenziano le caratteristiche salienti da modellare nella successiva fase della progettazione concettuale. Nella modulistica prodotta vengono evidenziati i fatti, le misure e le dimensioni coinvolte.

Dai requisiti di analisi raccolti in precedenza (Sottosezione 2.4.1) discende la seguente specifica dei requisiti del fatto *Registrazione delle ore* che associa ad ogni requisito di analisi

le dimensioni, le misure e le aggregazioni coinvolte. È importante specificare, in tale contesto, che per *FTE* si intende il valore ottenuto dalla seguente formula:

$$FTE: ActualWorkBillable / FTE_Hours$$

In cui *ActualWorkBillable* indica il numero di ore lavorative fatturabili registrate e *FTE_Hours* il numero di ore lavorative settimanali del dipendente. Se *ActualWorkBillable* viene considerata una misura, *FTE_Hours* è un attributo numerico statico associato a ogni dipendente.

Per *TurnoverActual*, invece, si intende il valore ottenuto dalla seguente formula:

$$TurnoverActual = ActualWorkBillable * ResourceStandardRate$$

In cui *ActualWorkBillable* indica sempre il numero di ore lavorative fatturabili registrate, mentre per *ResourceStandardRate* si intende il costo del lavoro, standardizzato, associato a ogni singolo dipendente.

Fatto Registrazione delle ore

N.	Requisito di analisi	Dimensioni	Misure preliminari	Aggregazioni
1	Numero di FTE associati a progetti il cui WBS Type corrisponde a <i>Base Service</i> , svolti da risorse operanti <i>On-Site</i> , per un dato livello della gerarchia organizzativa e per data.	Date, Resource, Organization, Projects, WBSType	ActualWorkBillable	SUM(FTE)
2	Numero di FTE associati a progetti il cui WBS Type corrisponde a <i>Base Service</i> , svolti da risorse operanti <i>Off-Shore</i> , per un dato livello della gerarchia organizzativa e per data.	Date, Resource, Organization, Projects, WBSType	ActualWorkBillable	SUM(FTE)

3	Numero di FTE associati a progetti il cui WBS Type corrisponde a <i>Minor Enhancements</i> , svolti da risorse operanti <i>On-Site</i> , per un dato livello della gerarchia organizzativa e per data.	Date, Resource, Organization, Projects, WBSType	ActualWorkBillable	SUM(FTE)
4	Numero di FTE associati a progetti il cui WBS Type corrisponde a <i>Minor Enhancements</i> , svolti da risorse operanti <i>Off-Shore</i> , per un dato livello della gerarchia organizzativa e per data.	Date, Resource, Organization, Projects, WBSType	ActualWorkBillable	SUM(FTE)
5	Costo medio orario relativo a progetti il cui WBS Type corrisponde a <i>Minor Enhancements</i> per un dato livello della gerarchia organizzativa e per data.	Date, Resource, Organization, Projects, WBSType	ActualWork Billable	SUM (Turnover Actual) / SUM (Actual Work Billable)
6	Numero di FTE associati a task di tipo <i>Unassigned time</i> , per un dato livello della gerarchia organizzativa e per data.	Date, Resource, Organization, Task	ActualWorkBillable	SUM(FTE)
7	Numero di FTE associati a progetti il cui WBS Type corrisponde a <i>Internal time</i> , per un dato livello della gerarchia organizzativa e	Date, Resource, Organization, Projects, WBSType	ActualWorkBillable	SUM(FTE)

	per data.			
8	Numero di FTE associati a progetti svolti da risorse operanti <i>On-Site</i> , per un dato livello della gerarchia organizzativa e per data.	Date, Resource, Organization, Projects.	ActualWorkBillable	SUM(FTE)
9	Numero di FTE associati a progetti svolti da risorse operanti <i>Off-Shore</i> , per un dato livello della gerarchia organizzativa e per data.	Date, Resource, Organization, Projects.	ActualWorkBillable	SUM(FTE)
10	Percentuale di lavoro associata a progetti il cui WBS Type corrisponde a <i>Minor Enhancements</i> svolti da risorse operanti <i>Off-Shore</i> , per un dato livello della gerarchia organizzativa e per data.	Date, Resource, Organization, Projects, WBSType	ActualWorkBillable	$\frac{\text{SUM(FTE KPI9)}}{\text{SUM(FTEKPI9)+SUM(FTE KPI8)}}$
11	Numero di ore lavorative fatturabili associate a progetti il cui WBS Type corrisponde a <i>Base Service</i> e task di tipo <i>Unassigned time</i> , per un dato livello della gerarchia organizzativa e per data.	Date, Resource, Organization, Projects, Task, WBSType	ActualWorkBillable	SUM(Actual Work Billable)

Si descrive il fatto *Registrazione delle ore*, specificandone la granularità, e quindi il suo significato, le dimensioni e le misure preliminari interessate.

La granularità del singolo fatto determina la dimensione del *data mart* e il tipo di analisi che si può effettuare sui dati; come suggerito da [Albano 12], in generale è preferibile scegliere una

granularità fina in modo da rendere possibili eventuali analisi dettagliate. Sebbene, come discusso in precedenza, i manager non siano interessati alla visualizzazione degli indicatori per i singoli dipendenti ma solo per i membri della gerarchia organizzativa, per effettuare correttamente i calcoli è necessario affinare la granularità del fatto a livello della singola risorsa umana. Inoltre, in fase di riunione preliminare, è emersa la volontà da parte del committente di estendere l'analisi di alcuni indicatori al livello del singolo dipendente come possibile sviluppo futuro.

Fatto Registrazione delle ore

Descrizione	Dimensioni Preliminari	Misure Preliminari
Un fatto riguarda la registrazione delle ore di lavoro svolte in una giornata da un dipendente appartenente a un Subteam in relazione a uno specifico task o a un progetto riguardante una determinata categoria.	Resource, Organization, Project, WBSType, Task, Date.	Actual Work Billable

Si descrivono le dimensioni del fatto *Registrazione delle ore* specificando per ognuna di esse il nome, una descrizione e la granularità.

Dimensioni

Nome	Descrizione	Granularità
Resource	La singola risorsa umana con tutte le proprietà ad essa associate.	Una risorsa umana.
Organization	La gerarchia organizzativa Accenture.	Un Subteam.
Projects	I progetti svolti dai dipendenti.	Un progetto.
Task	I task svolti dai dipendenti.	Un task.

Date	Dimensione temporale in cui si svolgono i fatti.	Un giorno.
WBS Type	Categorie standard utilizzate per catalogare le attività svolte.	Una tipologia WBS.

Per ogni dimensione vengono elencati gli attributi e una loro descrizione. L'elenco è completo di tutti gli attributi ritenuti rilevanti in fase di analisi, inclusi alcuni attributi non coinvolti direttamente nella raccolta requisiti ma ritenuti, di comune accordo con il committente, potenzialmente utili per eventuali analisi future.

Resource

Attributo	Descrizione
EmployeeName	Nominativo dell'impiegato.
EnterpriseID	Identificativo dell'impiegato all'interno del sistema telematico Accenture (in generale corrisponde a nome.cognome).
HomeOffice	Nominativo della sede di appartenenza (ad esempio ON-SHORE.Kolding).
Subteam_Code	Codice del Subteam al quale l'impiegato è assegnato.
Location_Name	Nome generico utilizzato per distinguere i dipendenti che lavorano in Danimarca da quelli che lavorano nelle altre sedi distaccate. Assume i valori On-Shore e Off-Shore.
FTE Hours	Valore numerico che indica il numero di ore lavorative settimanali del dipendente (assume i valori 40 e 45).
ResourceStandardRate	Costo del lavoro, standardizzato, associato a ogni singolo dipendente.
FTE Hours_Code	Codice delle ore lavorative settimanali.
Velux_ID	Identificativo dell'impiegato all'interno del

	sistema Velux (è presente solo per i dipendenti Velux).
RowIsCurrent	Valore binario utilizzato per gestire il cambiamento. Assume il valore 1 se la riga è attuale, 0 se fa riferimento a delle informazioni storiche.
RowValidFrom	Data che indica, sempre in riferimento alla gestione del cambiamento, il giorno a partire dal quale le informazioni assumono validità.
RowValidTo	Data che indica, sempre in riferimento alla gestione del cambiamento, il giorno in cui le informazioni cessano di essere attuali.

Projects

Attributo	Descrizione
ProjectName	Nominativo del progetto.
WBSTypeCode	Codice WBS associato al progetto.

Organization

Attributo	Descrizione
Organization	L'Organizzazione.
Tower	Il Tower Level.
Team	Il Team.
Subteam	Il Subteam.
RowIsCurrent	Valore binario utilizzato per gestire il cambiamento. Assume il valore 1 se la riga è attuale, 0 se fa riferimento a delle informazioni storiche.
RowValidFrom	Data che indica, sempre in riferimento alla gestione del cambiamento, il giorno a partire dal

	quale le informazioni assumono validità.
RowValidTo	Data che indica, sempre in riferimento alla gestione del cambiamento, il giorno in cui le informazioni cessano di essere attuali.

Task

Attributo	Descrizione
TaskGroupName	Nominativo del Gruppo a cui il Task appartiene.
TaskName	Nominativo del Task.
ME_Number	Campo che indica, nel caso in cui il task sia associato a un ticket, il codice del ticket.

WBS Type

Attributo	Descrizione
WBS_Name	Nominativo del WBS Type.

Date

Attributo	Descrizione
Day	Giorno
Week	Settimana
Month	Mese
Quarter	Trimestre
Year	Anno

Si descrivono le gerarchie dimensionali specificando per ogni dimensione le possibili gerarchie fra gli attributi e il loro tipo (bilanciata, incompleta, ricorsiva). La dimensione viene omessa se non presenta alcuna gerarchia.

Gerarchie dimensionali

Dimensione	Descrizione	Tipo di Gerarchia
Organization	Subteam → Team → Tower → Organization	Bilanciata
Date	Day → Month → Quarter → Year	Bilanciata
Date	Week → Year	Bilanciata

Si specifica il tipo di strategia da usare per trattare le dimensioni con attributi che possono cambiare nel tempo. Si elenca quindi la lista delle dimensioni ricavate, evidenziandone la granularità e la metodologia da utilizzare per il trattamento delle modifiche.

Le tipologie di trattamento delle modifiche possibili sono quattro, delle quali le prime tre si considerano quando qualche attributo dimensionale cambia raramente (*slowly changing dimensions*) [Albano 12]:

- **Perdita di storia (Tipo 1).** Il valore dell'attributo dimensionale che cambia viene sostituito con il nuovo valore. È la soluzione più semplice ed immediata, ma si perde la possibilità di storicizzare i cambiamenti.
- **Storicizzazione (Tipo 2).** Si aggiunge una nuova riga alla tabella dimensionale, creando di fatto una entità nuova. Tutti i fatti precedenti alla modifica fanno riferimento alla vecchia entità, mentre tutti i fatti successivi alla modifica fanno riferimento a quella nuova. In questo modo il caricamento dei dati si complica, aumentano i dati della dimensione, ma si ha la possibilità di storicizzare il cambiamento.
- **Storicizzazione con data cambiamento (Tipo 3).** Con questo trattamento si memorizza, oltre che alla storia, anche il momento temporale in cui avviene il cambiamento. Per ottenere questo risultato è necessario sostituire l'Attributo con tre campi: Attributo, Nuovo_Attributo, Data_Modifica.

- **Elevata frequenza di cambiamento (Tipo 4).** Per gli attributi dimensionali che cambiano molto frequentemente si possono prevedere due tabelle dimensionali, una contenente gli attributi che rimangono immutati e una contenente gli attributi che variano.

Di seguito viene descritta la modalità di trattamento delle dimensioni coinvolte nel fatto *Registrazione delle ore*. I dettagli sulla strategia adottata per il trattamento delle modifiche verranno approfonditi nella Sezione 6.8.

Nome	Granularità	Trattamento modifiche
Resource	La risorsa	Tipo 2
Organization	Il Subteam	Tipo 2
Projects	Il progetto	Tipo 1
Task	Il task	Tipo 1
WBS Type	La tipologia	Tipo 1
Date	Il giorno	Tipo 1

Si descrive ogni misura del fatto *Registrazione delle ore*, come derivarla da altre misure, se calcolata, e il tipo di aggregabilità, ovvero quali funzioni di aggregazione sono applicabili alla misura quando si raggruppano i dati secondo certe dimensioni.

Misure

Misura	Descrizione	Aggregabilità	Calcolata
Actual Work Billable	Il numero di ore di lavoro fatturabile registrate dal dipendente.	Additiva	NO

3.2.2 Progettazione concettuale iniziale del data mart

Si mostra di seguito lo schema concettuale iniziale del *data mart* Registrazione delle ore, suggerito dall'analisi dei requisiti (Figura 3.2). Lo schema è una rappresentazione, con il modello dimensionale dei fatti, dei requisiti di analisi trovati.

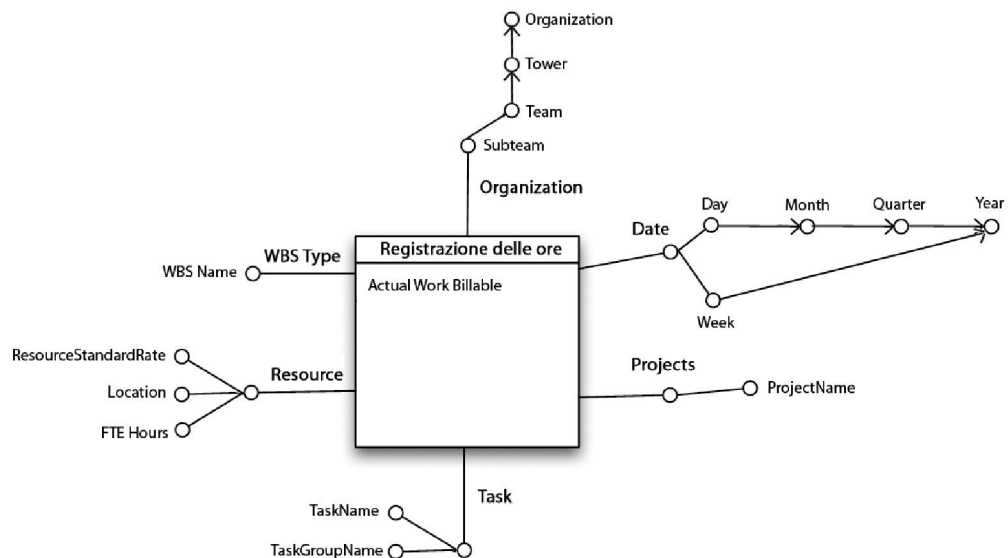


Figura 3.2: Schema concettuale iniziale del *data mart* Registrazione delle ore.

3.3 Ticket di servizio

Si descrive la specifica dei requisiti e la progettazione concettuale iniziale del *data mart* *Ticket di servizio*. Il fatto riguarda lo stato evolutivo di un ticket caratterizzato da determinate proprietà, la cui risoluzione è affidata a un dipendente appartenente a un livello della gerarchia organizzativa e relativo a un task in un determinato istante di tempo.

3.3.1 Specifica dei requisiti del fatto

Dai requisiti di analisi raccolti in precedenza (Sottosezione 2.4.2) discende la seguente specifica dei requisiti del fatto *Ticket di servizio* che associa ad ogni requisito di analisi le

dimensioni, le misure e le aggregazioni coinvolte. La notazione *COUNT/COUNT(*)*, la quale è presente in alcune tra le aggregazioni che seguono, è volta a indicare che al numeratore vengono conteggiati i record appartenenti a una determinata tipologia (ad esempio, nel caso del requisito numero 12, gli *Incidents*) che soddisfano il requisito di analisi, mentre al denominatore viene conteggiata la totalità dei record appartenenti alla stessa tipologia, sempre in riferimento a un determinato periodo di tempo.

Fatto Ticket di servizio

N.	Requisito di analisi	Dimensioni	Misure	Aggregazioni
12	Percentuale di ticket di tipo Incident risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Date, TicketDetails, TicketStatus.		COUNT/ COUNT(*)
13	Numero di ticket di tipo Incident non risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Date, TicketDetails, TicketStatus.		COUNT
14	Numero medio di giorni di vita dei ticket di tipo Incident che non sono stati risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Date, TicketDetails, TicketStatus.		AVG
15	Percentuale di ticket di tipo Request risolti entro i termini prestabiliti in	Resource, Organization, Task,		COUNT/

	referimento, per un dato livello della gerarchia organizzativa e per data.	Date, TicketDetails, TicketStatus.		COUNT(*)
16	Numero di ticket di tipo Request non risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Task, Date, TicketDetails, TicketStatus.		COUNT
17	Numero medio di giorni di vita dei ticket di tipo Request che non sono stati risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Task, Date, TicketDetails, TicketStatus.		AVG
18	Numero medio di ore spese per risolvere un ticket di tipo Request, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Task, Date, TicketDetails, TicketStatus.	TimeSpend On MEs	AVG(TimeSpend On MEs)
19	Numero di ore preventivate per la risoluzione dei ticket di tipo Request, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Task, Date, TicketDetails, TicketStatus.	TimeSpend On MEs	SUM (TimeSpend On MEs)
20	Percentuale di ticket di tipo Request che sono stati risolti in un numero di ore	Resource, Organization, Task,		

	maggiore rispetto a quelle preventivate, per un dato livello della gerarchia organizzativa e per data.	Date, TicketDetails, TicketStatus.	TimeSpend On MEs	COUNT/ COUNT(*)
21	Numero medio di ore in eccesso dei ticket di tipo Request che sono stati risolti in un numero di ore maggiore rispetto a quelle preventivate, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Task, Date, TicketDetails, TicketStatus.	TimeSpend On MEs	AVG(TimeSpend On MEs)
22	Numero di ticket di tipo Request la cui data relativa all'ultimo aggiornamento eccede i 30 giorni, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Task, Date, TicketDetails, TicketStatus.		COUNT
23	Numero di ticket di tipo Incident la cui data relativa all'ultimo aggiornamento eccede i 7 giorni, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Date, TicketDetails, TicketStatus.		COUNT
24	Numero di ticket di tipo Problem marcati come Open, per un dato livello della gerarchia	Resource, Organization, Date, TicketDetails,		COUNT

	organizzativa e per data.	TicketStatus.		
25	Numero di ticket di tipo Problem non risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Date, TicketDetails, TicketStatus.		COUNT
26	Percentuale di ticket di tipo Incident associati a una particolare categoria di persone definita Critical User risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Date, TicketDetails, TicketStatus.		COUNT/ COUNT(*)
27	Percentuale di ticket di tipo Request associati a una particolare categoria di persone definita Critical User risolti entro i termini prestabiliti, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Task, Date, TicketDetails, TicketStatus.		COUNT/ COUNT(*)
28	Numero assoluto di ticket di tipo Request che sono stati risolti in un numero di ore maggiore rispetto a quelle preventivate, per un dato livello della gerarchia organizzativa e per data.	Resource, Organization, Task, Date, TicketDetails, TicketStatus.	TimeSpend On MEs	COUNT

Si descrive il fatto *Ticket di servizio*, specificandone la granularità, e quindi il suo significato, le dimensioni e le misure preliminari interessate. Per quanto riguarda la granularità vale il concetto enunciato nella Sottosezione 3.2.1.

Fatto Ticket di Servizio

Descrizione	Dimensioni Preliminari	Misure Preliminari
Un fatto riguarda lo stato evolutivo di un ticket caratterizzato da determinate proprietà, la cui risoluzione è affidata a un dipendente appartenente a un livello della gerarchia organizzativa e relativo a un task in un determinato istante di tempo.	Resource, Organization, Task, Date, TicketDetails, TicketStatus.	TimeSpendOnMEs.

Si descrivono le dimensioni del fatto Ticket di servizio specificando per ognuna di esse il nome, una descrizione e la granularità.

Dimensioni

Nome	Descrizione	Granularità
Resource	La singola risorsa umana con tutte le proprietà ad essa associate.	Una risorsa umana.
Organization	La gerarchia organizzativa Accenture.	Un membro della gerarchia organizzativa.
Task	I task svolti dai dipendenti.	Un task.
Date	Dimensione temporale in cui si svolgono i fatti.	Un giorno.
TicketDetails	I dettagli del singolo ticket.	Un ticket.
TicketStatus	Lo status del proprio ciclo di vita in cui si trova il ticket.	Uno status.

Per ogni dimensione vengono elencati gli attributi e una loro descrizione, a volte comprensiva di formato.

TicketDetails

Attributo	Descrizione
TicketCode	Codice del ticket.
PlannedEnd	Termine entro il quale si prevede di risolvere il ticket.
CriticalUser	Valore binario che indica se il ticket è associato a una particolare categoria di persone definita <i>Critical User</i> .
Category	Valore testuale che indica a quale categoria fa riferimento il ticket (ad esempio SOFTWARE o PROCESSES).
Subcategory	Valore testuale che specifica in maniera più dettagliata a quale sottocategoria fa riferimento il ticket (ad esempio MOBILE o SERVICE MANAGEMENT).
TicketType	Tipologia del ticket (Incident, Request, Problem).
RequestType	Valore che indica, nel caso il ticket sia di tipo “Request”, a quale tipo di richiesta faccia riferimento (Project, Base Service, Purchase Minor Enhancement o Fast Procurement).
EstimatedHours	Numero di ore stimate per la risoluzione del ticket.

TicketStatus

Attributo	Descrizione
TicketStatus	Indica lo status in cui si trova il ticket.
ReportingStatus	Indica lo status in cui si trova il ticket in

	funzione dei valori interessanti da visualizzare nei report (Open, Resolved, Closed)
--	--

Resource

Attributo	Descrizione
EmployeeName	Nominativo dell'impiegato.
EnterpriseID	Identificativo dell'impiegato all'interno del sistema telematico Accenture (in generale corrisponde a nome.cognome).
HomeOffice	Nominativo della sede di appartenenza (ad esempio ON-SHORE.Kolding).
Subteam_Code	Codice del Subteam al quale l'impiegato è assegnato.
Location_Name	Nome generico utilizzato per distinguere i dipendenti che lavorano in Danimarca da quelli che lavorano nelle altre sedi distaccate. Assume i valori On-Shore e Off-Shore.
FTE Hours	Valore numerico che indica il numero di ore lavorative settimanali del dipendente (assume i valori 40 e 45).
ResourceStandardRate	Costo del lavoro, standardizzato, associato a ogni singolo dipendente.
FTE Hours_Code	Codice delle ore lavorative settimanali.
Velux_ID	Identificativo dell'impiegato all'interno del sistema Velux (è presente solo per i dipendenti Velux).
RowIsCurrent	Valore binario utilizzato per gestire il cambiamento. Assume il valore 1 se la riga è attuale, 0 se fa riferimento a delle informazioni storiche.
RowValidFrom	Data che indica, sempre in riferimento alla

	gestione del cambiamento, il giorno a partire dal quale le informazioni assumono validità.
RowValidTo	Data che indica, sempre in riferimento alla gestione del cambiamento, il giorno in cui le informazioni cessano di essere attuali.

Organization

Attributo	Descrizione
Organization	L'Organizzazione.
Tower	Il Tower Level.
Team	Il Team.
Subteam	Il Subteam.
RowIsCurrent	Valore binario utilizzato per gestire il cambiamento. Assume il valore 1 se la riga è attuale, 0 se fa riferimento a delle informazioni storiche.
RowValidFrom	Data che indica, sempre in riferimento alla gestione del cambiamento, il giorno a partire dal quale le informazioni assumono validità.
RowValidTo	Data che indica, sempre in riferimento alla gestione del cambiamento, il giorno in cui le informazioni cessano di essere attuali.

Task

Attributo	Descrizione
TaskGroupName	Nominativo del Gruppo a cui il Task appartiene.
TaskName	Nominativo del Task.
ME_Number	Campo che indica, nel caso in cui il task sia associato a un ticket, il codice del ticket.

Date

Attributo	Descrizione
Day	Giorno
Week	Settimana
Month	Mese
Quarter	Trimestre
Year	Anno

Si descrivono le gerarchie dimensionali specificando per ogni dimensione le possibili gerarchie fra gli attributi e il loro tipo (bilanciata, incompleta, ricorsiva). La dimensione viene omessa se non presenta alcuna gerarchia.

Gerarchie dimensionali

Dimensione	Descrizione	Tipo di Gerarchia
Organization	Subteam → Team → Tower → Organization	Bilanciata
Date	Day → Month → Quarter → Year	Bilanciata
Date	Week → Year	Bilanciata

Viene descritta la modalità di trattamento delle dimensioni coinvolte nel fatto Ticket di Servizio.

Nome	Granularità	Trattamento modifiche
Resource	La risorsa	Tipo 2
Organization	Il Subteam	Tipo 2
Task	Il task	Tipo 1
Date	Il giorno	Tipo 1
TicketDetails	Il ticket	Tipo 1

Si descrive ogni misura del fatto Ticket di servizio, come derivarla da altre misure, se calcolata, e il tipo di aggregabilità, ovvero quali funzioni di aggregazione sono applicabili alla misura quando si raggruppano i dati secondo certe dimensioni.

Misure

Misura	Descrizione	Aggregabilità	Calcolata
TimeSpendOnMEs	Numero di ore spese per risolvere ticket di tipo Request.	Additiva	SI

3.3.2 Progettazione concettuale iniziale del data mart

Si mostra di seguito lo schema concettuale iniziale del *data mart* Ticket di servizio, suggerito dall'analisi dei requisiti (Figura 3.3). Lo schema è una rappresentazione, con il modello dimensionale dei fatti, dei requisiti di analisi trovati.

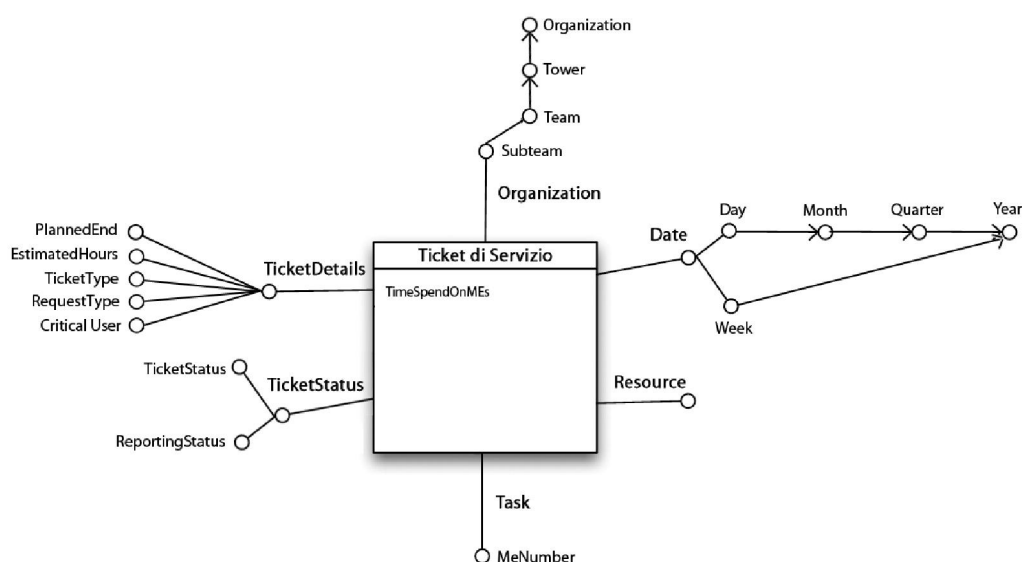


Figura 3.3: Schema concettuale iniziale del data mart Ticket di servizio

3.4 Tabelle riepilogative delle dimensioni e delle misure

Prima di procedere con la progettazione concettuale a partire dai dati operazionali, la redazione dei *data mart* finali e la conclusiva progettazione logica del *data warehouse*, viene presentata una tabella di riepilogo che illustra quali sono le dimensioni e le misure condivise da più fatti.

Dimensioni dei fatti

Dimensione	Registrazione delle ore	Ticket di Servizio
Resource	X	X
Project	X	
Organization	X	X
Task	X	X
WBSType	X	
Date	X	X
TicketDetails		X
TicketStatus		X

Misure dei fatti

Misura	Registrazione delle ore	Ticket di Servizio
Actual Work Billable	X	
TimeSpendOnMEs		X

PROGETTAZIONE CONCETTUALE FINALE E LOGICA DEI *DATA MART* E DEL *DATA WAREHOUSE*

In questo capitolo vengono brevemente presentati sia i sistemi che accolgono le basi di dati operazionali, sia le caratteristiche delle basi di dati considerate nell'analisi. Dopo l'analisi sulle tabelle e i campi necessari alla realizzazione del *data warehouse*, vengono definite le entità (evento, componente e classificazione) e gli schemi concettuali finali dei *data mart*. Infine vengono presentati gli schemi logici dei singoli *data mart* e del *data warehouse*.

4.1 Introduzione ai sistemi sorgenti

“Ogni organizzazione per offrire i propri servizi utilizza una base di dati operazionale (o transazionale) che contiene i dati necessari per svolgere le operazioni giornaliere di routine” [Albano 12]. Come evidenzia [Albano 12], la base di dati operazionale è usata tramite opportune applicazioni che fanno parte dei cosiddetti sistemi informatici operazionali o sistemi di elaborazione di transazioni (Transaction Processing Systems, TPS). Le applicazioni consentono agli utenti del livello operativo di svolgere in modo corretto le attività di loro competenza (Figura 4.1).

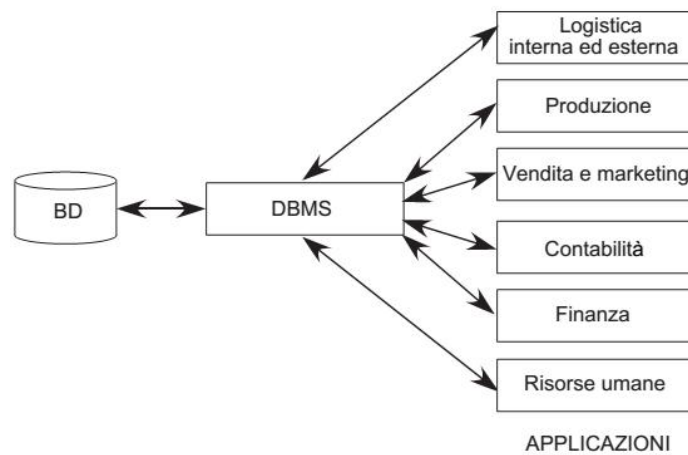


Figura 4.1: Sistemi informatici operazionali [Albano 12].

I sistemi informatici operazionali coinvolti nel processo di realizzazione del *data warehouse* sono molteplici e affinché quest'ultimo possa essere realizzato in linea con la modellazione presentata nel capitolo 2, i dati da essi provenienti devono essere tra loro integrati. In Figura 4.2 è mostrato uno schema riassuntivo del flusso di dati, mentre le procedure di selezione delle tabelle e degli attributi sono descritte nei paragrafi a seguire.

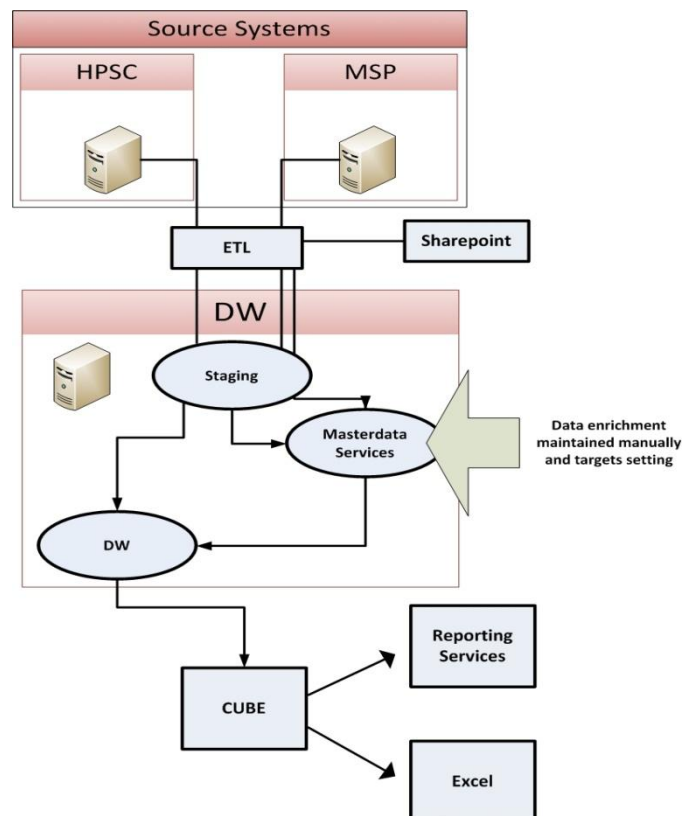


Figura 4.2: Sistemi sorgenti e flusso di dati

In particolare viene mostrato come i dati presenti nei due sistemi sorgenti debbano essere integrati con i dati estratti da Accenture SharePoint prima di essere elaborati tramite procedure di ETL e con i dati inseriti e sottoposti a revisione tramite Master Data Services prima di popolare dimensioni e fatti. La parte finale della Figura 4.2 mostra come i dati presenti nei report vengano ricavati tramite una connessione diretta con il cubo.

4.2 Presentazione della base di dati operativa MSP

Accenture prevede un rigido iter burocratico da seguire per poter ottenere l'accesso ai propri dati. È stato necessario prendere visione del regolamento sul trattamento dei dati, sottoscriverlo, sottoporsi a un questionario online e superarlo per poter ottenere la creazione di un account col quale accedere al server MSP. Accenture ha concesso in dotazione un generatore di password, che congiuntamente all'inserimento delle credenziali, permetteva l'accesso, tramite VPN (Virtual Private Network), a parte della base di dati operativa (circa un centinaio di tabelle) contenente le informazioni sulle attività del reparto ADU. In prima analisi la base di dati è parsa confusionaria, con tabelle e dati ridondanti ed è stato tutt'altro che intuitivo comprendere quali selezionare e quali scartare. D'altro canto, solo una piccola porzione della base di dati è significativa per la nostra analisi e rappresentarla in modo completo genererebbe solo confusione. Sulla base dei requisiti di analisi dei dati, si esamina lo schema relazionale per decidere quali tabelle e attributi sono interessanti. Lo schema relazionale che è di seguito rappresentato (Figura 4.3) contiene dunque solo le tabelle importanti (delle quali vengono indicati solo i campi più significativi) ai fini del disegno dei *data mart* candidati.

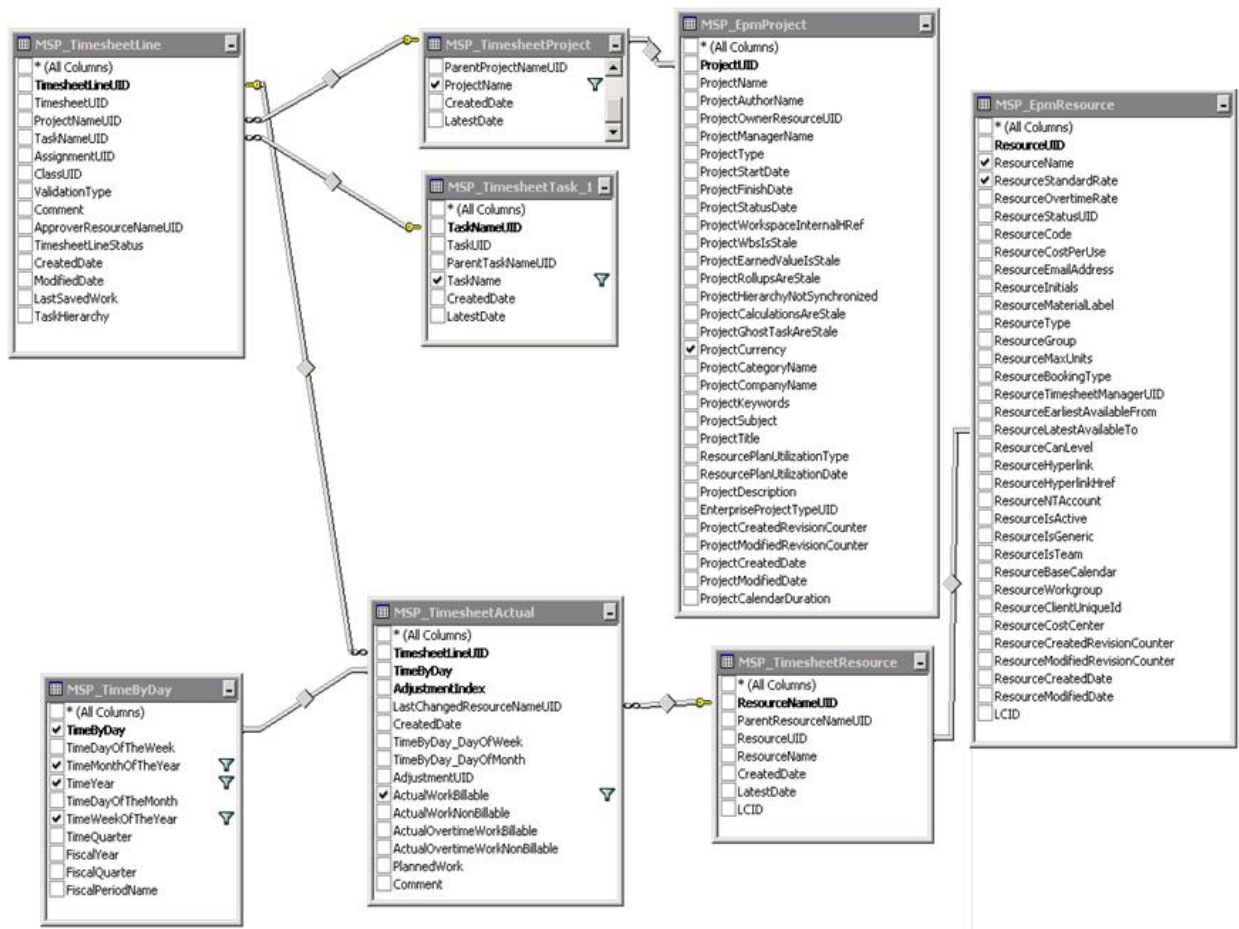


Figura 4.3: Base di dati operativa parziale MSP.

Viene descritto sinteticamente il contenuto di ognuna delle tabelle indicate nello schema della base di dati:

- MSP_TimesheetActual contiene le registrazioni delle ore di lavoro;
- MSP_TimesheetLine è una tabella intermedia che collega la precedente tabella alle tabelle che seguono;
- MSP_TimesheetProject contiene la chiave dei progetti;
- MSP_TimesheetTask contiene le informazioni rilevanti sul task;
- MSP_EpmProjects contiene alcuni dettagli sui progetti;
- MSP_TimesheetResource è una tabella intermedia che collega MSP_TimesheetActual a MSP_TimesheetEpmResource;
- MSP_TimesheetEpmResource contiene alcuni dettagli sulla risorsa umana.

A questo punto, come suggerito da [Albano 12], si classificano le tabelle interessanti in base al loro contenuto ed alle associazioni fra di loro, come entità evento, entità componente e entità di classificazione.

- **Entità Evento.** Sono le tabelle che rappresentano eventi potenzialmente interessanti per i processi aziendali di analisi dei dati. Le entità evento hanno due caratteristiche fondamentali: descrivono eventi che si verificano frequentemente nel tempo e contengono attributi numerici che rappresentano possibili misure. In questa categoria rientra la tabella `MSP_TimesheetActual`.
- **Entità Componente.** Sono le tabelle in relazione con un'entità evento con un'associazione (1:N). Le entità componente di solito rappresentano informazioni utili per rispondere a domande del tipo chi, cosa, quando, dove, come e perché relative alle entità evento e, quindi, sono le naturali candidate da considerare per la definizione delle dimensioni dello schema a stella. In questa categoria rientrano le tabelle `MSP_EpmProjects`, `MSP_TimesheetTask` e `MSP_TimesheetEpmResource`. Tra le entità componente di ogni entità evento aggiungiamo quella che rappresenta il tempo.
- **Entità di classificazione.** Sono le tabelle in relazione con un'entità componente (entità minimale) con una catena di associazioni (1:N). Queste entità rappresentano di solito gerarchie nei dati e, nella definizione dello schema a stella, i loro attributi interessanti vengono aggiunti a quelli delle entità minimali per definire le tabelle dimensionali. Non sono state rilevate tabelle di classificazione interessanti in questa base di dati. Per modellare la gerarchia della dimensione Organization è necessario integrare i dati provenienti da questa base di dati con i dati provenienti da altre fonti, Accenture SharePoint e Master Data Services.

4.3 MSP, classificazione delle tabelle interessanti

Prima di progettare i *data mart* candidati a partire dalla base di dati operativa mostrata in precedenza, bisogna evidenziare quali sono le tabelle utili per la costruzione degli schemi; per ogni tabella si selezionano gli attributi interessanti. Tutte le tabelle non riportate nello schema mostrato in precedenza devono considerarsi scartate, così come i campi delle tabelle riportate che sono stati omessi dalla rappresentazione. Allo stesso modo, si trascurano le chiavi delle

tabelle. Per quanto riguarda invece gli attributi delle altre tabelle, valgono le seguenti considerazioni:

- MSP_TimesheetActual: si estraggono Timebyday e ActualWorkBillable;
- MSP_TimesheetTask: si estrae TaskName;
- MSP_EpmProjects: si estrae ProjectName;
- MSP_TimesheetEpmResource: ResourceStandardRate e ResourceName.

4.4 Accenture Sharepoint e Master Data Services

I dati presenti nel server MSP non sono sufficienti per modellare le dimensioni Organization e Resource. Tali dati, infatti, sono ricavabili da un'ulteriore fonte esterna, denominata *Accenture SharePoint*, alla quale abbiamo avuto accesso parziale. Tra le varie tabelle presenti, sono state prese in considerazione:

- SP_Resources, la quale contiene i dettagli relativi ai singoli dipendenti;
- SP_Teams, la quale contiene i dettagli relativi ai team;
- SP_Subteams, la quale contiene i dettagli relativi ai sub team.

Per quanto riguarda invece gli attributi, valgono le seguenti considerazioni:

- SP_Resources: interessano EnterpriseID, HomeOffice, LocationName, FTEHours, FTEHoursCode, VeluxID;
- SP_Teams: interessano TeamName e TeamCode;
- SP_Subteams: interessano SubTeamName e SubTeamCode.

La strategia adottata è stata quella di estrarre i dati necessari alla modellazione una sola volta, renderli omogenei nel formato, importarli nella staging area (denominata *STAIMPORT*) e successivamente renderli disponibili alla modifica e al mapping manuale (l'associazione tra dipendenti e livelli gerarchici è difatti solamente parziale nello Sharepoint) tramite lo strumento *Microsoft Master Data Services* (d'ora in poi definito con l'acronimo MDS), del quale si parlerà in maniera approfondita nel quinto capitolo di questo documento. Tramite MDS sono state create le associazioni tra Team, Tower e Organization (questi ultimi due, essendo nel complesso un numero

ridotto, inseriti manualmente), nonché la tabella necessaria alla creazione della dimensione WBSType, dalla quale vengono poi estratti WBSName e WBSCode.

Una volta validati tramite MDS, i dati vengono raccolti e utilizzati per popolare le tabelle storiche tramite le quali si gestisce il cambiamento. Da questo punto in poi l'interfaccia con MDS è costante, e le tabelle storiche continuamente aggiornate e utilizzate come sorgente dapprima per creare le dimensioni e in seguito per mantenerle aggiornate.

4.5 Presentazione della base di dati operativa HPSC

Il processo di raccolta ed estrazione dati dal server HPSC è stato lungo e dispendioso a causa delle difficoltà incontrate per ottenere le tabelle necessarie alla modellazione. Nonostante ripetute sollecitazioni, durante tutta la durata del progetto Accenture non è riuscita a metterci in condizione di accedere direttamente alla base di dati. Le ragioni sono molteplici, ma la principale è legata al fatto che il server è gestito da personale operante nelle sedi asiatiche e le regole interne obbligano le diverse sedi a superare una lunga trafila affinché si possa giungere alla concessione dei permessi di accesso. Per poter superare questa fase di stallo e concludere la modellazione, uno dei gestori della base di dati ci ha fornito, staticamente, le tabelle necessarie contenenti un sottoinsieme dei record totali.

Viene descritto sinteticamente il contenuto di ognuna delle tabelle indicate nello schema della base di dati:

- RUBACTIVITY3M1 e OCMQM1 contengono i dati sui ticket di tipo Request;
- ACTIVITYM1 e PROBSUMMARYM1 contengono i dati sui ticket di tipo Incident;
- ROOTCAUSEM1 e ACTPRBM1 contengono dati sui ticket di tipo Problem;

4.6 HPSC, classificazione delle tabelle interessanti

Tutte le tabelle introdotte nella sezione precedente risultano utili per la costruzione degli schemi concettuali. Le tabelle contengono un elevato numero di campi non rilevanti ai fini della modellazione (la tabella con più campi, RUBACTIVITY3M1, ne contiene 284) e, sebbene contengano tutte lo stesso tipo di informazioni sui ticket, i formati dei dati risultano diversi tra loro (ad esempio, la data di apertura di un ticket è rappresentata in formati diversi da tabella a tabella). Per questo motivo, una volta identificati i dati da estrarre, è stato necessario renderli omogenei nel

formato tramite uno script SQL.

Gli attributi interessanti sono: TicketCode, OpenedTime, UpdatedTime, ResolvedTime, OpenedBy, UpdatedBy, ResolvedBy, ClosedBy, CriticalUser, PlannedEnd, Category, Subcategory, TicketType, RequestType, EstimatedHours, TicketStatus.

Gli attributi necessari alla modellazione non presenti nelle tabelle operazionali vengono ricavati in questo modo:

- L'attributo ReportingStatus della dimensione TicketStatus (che assume i valori Open, Resolved, Closed) viene invece ricavato dal più specifico TicketStatus.
- L'attributo MENumber della dimensione Task viene ricavato dall'attributo TaskName, dal momento che i task relativi a ticket di tipo Request presentano all'interno del proprio nominativo una stringa indicante il codice del ticket.

Infine, la misura dei fatti, TimeSpendOnMEs, ovvero il tempo speso nella risoluzione di ticket di tipo Request, viene ricavata dalla tabella MSP_TimesheetActual del server MSP tramite un'aggregazione sull'attributo ActualWorkBillable per mezzo dei task che corrispondono alla risoluzione di ticket di servizio.

4.7 KPI Targets

Un aspetto trascurato nella fase precedente ed emerso in quella corrente, anche grazie alla decisione di adottare MDS come strumento di data management, è la gestione dei cosiddetti valori “target” dei KPI, obiettivi da raggiungere a cadenze temporali nonché paragone di confronto per valutare la bontà dell'operato aziendale. Un target è pertanto un valore numerico al quale devono essere necessariamente associati il nominativo, il gruppo di appartenenza, l'unità di misura e l'andamento desiderato del rispettivo KPI. Il committente ha espresso la volontà di poter modificare agevolmente, a cadenze temporali, tutte queste caratteristiche. Tra le diverse opzioni di modellazione, è stato scelto di creare un *data mart* dedicato al processo di selezione dei target, considerando i KPI associati come una dimensione dei fatti. Le tabelle operazionali dalle quali vengono estratti i dati delle dimensioni e del fatto vengono create e gestite tramite MDS.

Fatto KPITargets

Descrizione	Dimensioni Preliminari	Misure Preliminari
Un fatto riguarda un valore target per uno specifico KPI in relazione a un livello della gerarchia organizzativa e un determinato periodo di tempo.	Organization, Date, KPIs.	TargetValue.

Si descrivono le dimensioni del fatto KPI Target specificando per ognuna di esse il nome, una descrizione e la granularità.

Dimensioni

Nome	Descrizione	Granularità
Organization	La gerarchia organizzativa Accenture.	Un membro della gerarchia organizzativa.
Date	Dimensione temporale in cui si svolgono i fatti.	Un giorno.
KPIs	I dettagli del singolo KPI.	Un KPI.

Per ogni dimensione vengono elencati gli attributi e una loro descrizione, a volte comprensiva di formato.

Organization

Attributo	Descrizione
Organization	L'Organizzazione.
Tower	Il Tower Level.
Team	Il Team.
Subteam	Il Subteam.
RowIsCurrent	Valore binario utilizzato per gestire il cambiamento. Assume il valore 1 se la riga è

	attuale, 0 se fa riferimento a delle informazioni storiche.
RowValidFrom	Data che indica, sempre in riferimento alla gestione del cambiamento, il giorno a partire dal quale le informazioni assumono validità.
RowValidTo	Data che indica, sempre in riferimento alla gestione del cambiamento, il giorno in cui le informazioni cessano di essere attuali.

KPIs

Attributo	Descrizione
Name	Nome del KPI.
Group	Nome del gruppo a cui appartiene il KPI.
Unit	Unità di misura del KPI.
DesiredTrend	Trend desiderato (positivo, negativo).

Date

Attributo	Descrizione
Day	Giorno
Week	Settimana
Month	Mese
Quarter	Trimestre
Year	Anno

Si descrivono le gerarchie dimensionali specificando per ogni dimensione le possibili gerarchie fra gli attributi e il loro tipo (bilanciata, incompleta, ricorsiva). La dimensione viene omessa se non presenta alcuna gerarchia.

Gerarchie dimensionali

Dimensione	Descrizione	Tipo di Gerarchia
Organization	Subteam → Team → Tower → Organization	Bilanciata
Date	Day → Month → Quarter → Year	Bilanciata
Date	Week → Year	Bilanciata

Viene descritta la modalità di trattamento delle dimensioni coinvolte nel fatto Registrazione delle ore.

Nome	Granularità	Trattamento modifiche
Organization	Il Subteam	Tipo 2
KPIs	Il KPI	Tipo 1
Date	Il giorno	Tipo 1

Si descrive ogni misura del fatto KPI Targets, come derivarla da altre misure, se calcolata, e il tipo di aggregabilità, ovvero quali funzioni di aggregazione sono applicabili alla misura quando si raggruppano i dati secondo certe dimensioni.

Misure

Misura	Descrizione	Aggregabilità	Calcolata
TargetValue	Valore di riferimento per il KPI.	Semi additiva	NO

4.7.1 Progettazione concettuale iniziale del data mart

Si mostra di seguito lo schema concettuale iniziale del *data mart* KPI Targets, suggerito dall'analisi dei requisiti (Figura 4.4). Lo schema è una rappresentazione, con il modello dimensionale dei fatti, dei requisiti di analisi trovati.

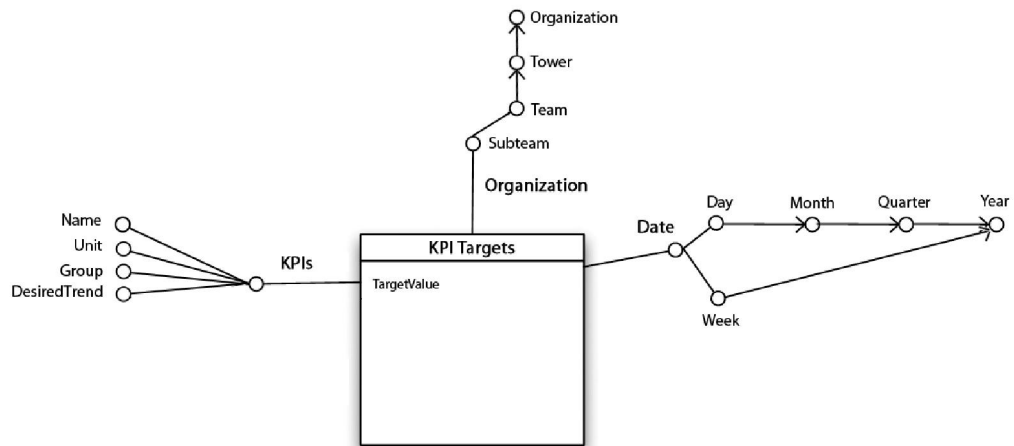


Figura 4.4: Schema concettuale iniziale del *data mart* KPI Targets

4.8 Progettazione logica del *data mart* Registrazione delle ore

Prima di mostrare lo schema logico del *data mart* Registrazione delle ore è bene ricordare i passi salienti della progettazione logica dei *data mart* e del *data warehouse*. Come proposto da [Albano 12], in questa fase si passa dagli schemi finali dei *data mart* (che nel nostro caso coincidono con quelli iniziali dal momento che sono stati reperiti tutti i dati necessari) allo schema relazionale del *data warehouse*, trasformando prima i tre schemi concettuali dei *data mart* nei quattro schemi relazionali a stella, e poi integrando i vari schemi dei *data mart* in un unico schema del *data warehouse*. In Figura 4.5 è mostrato lo schema logico del *data mart* Registrazione delle ore.

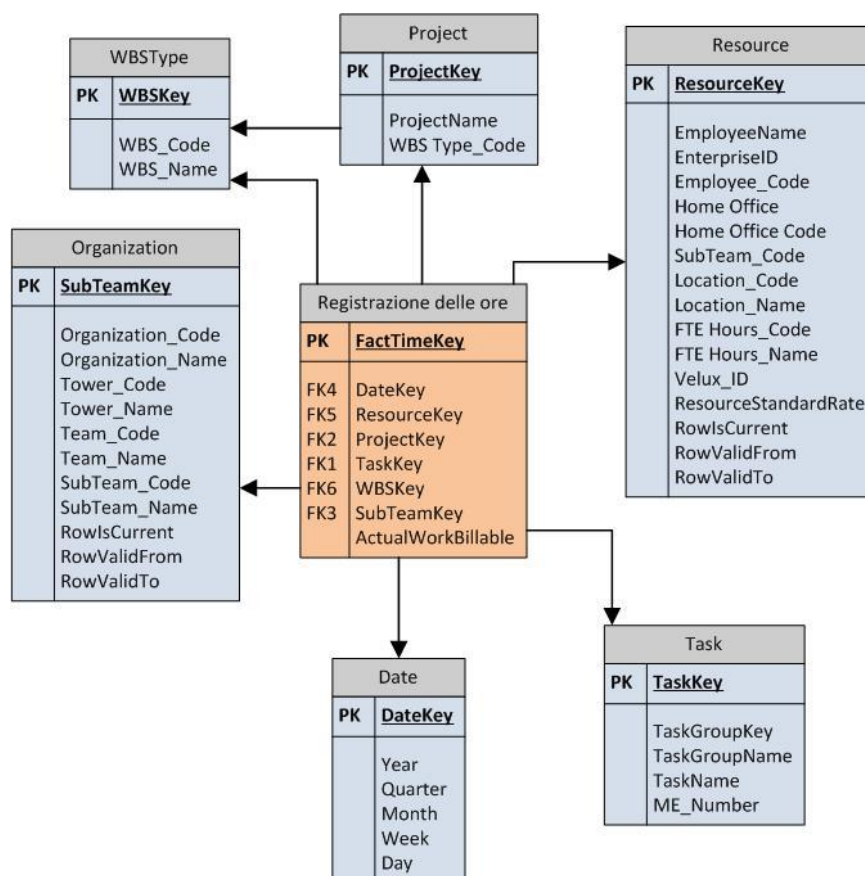


Figura 4.5: Schema logico del *data mart* Registrazione delle ore

4.9 Progettazione logica del *data mart* Ticket di servizio

In Figura 4.6 è mostrato lo schema logico del *data mart* Ticket di servizio.

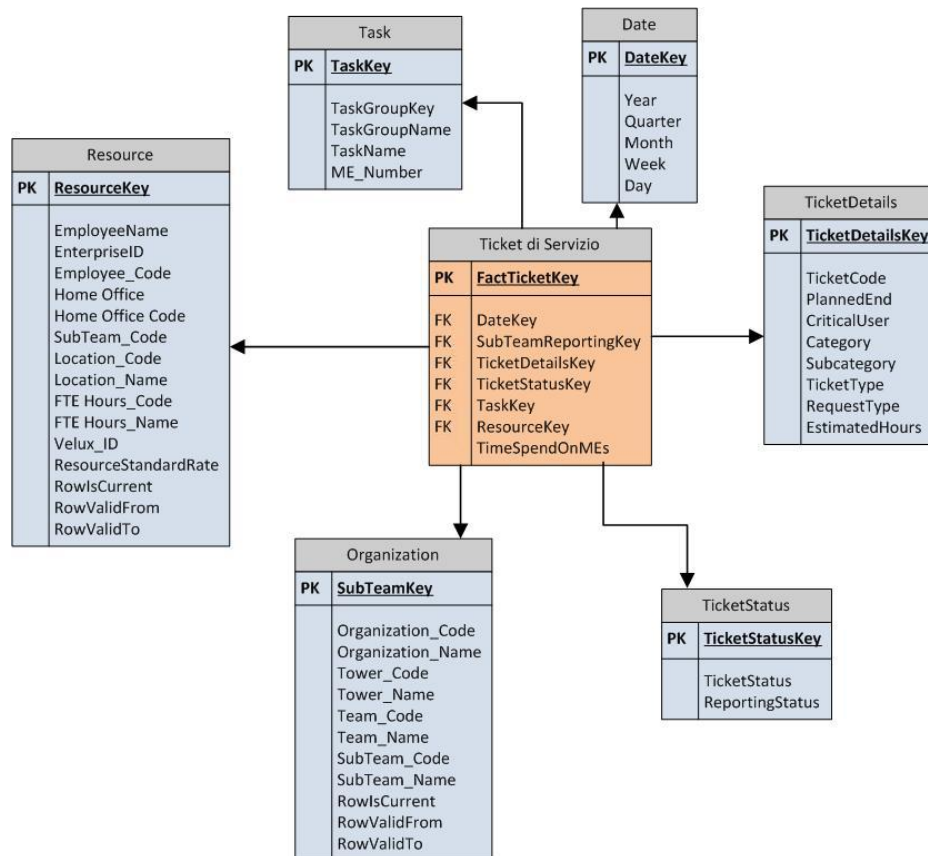


Figura 4.6: Schema logico del *data mart* Ticket di servizio

4.10 Progettazione logica del *data mart* KPI Targets

In Figura 4.7 è mostrato lo schema logico del *data mart* KPITargets.

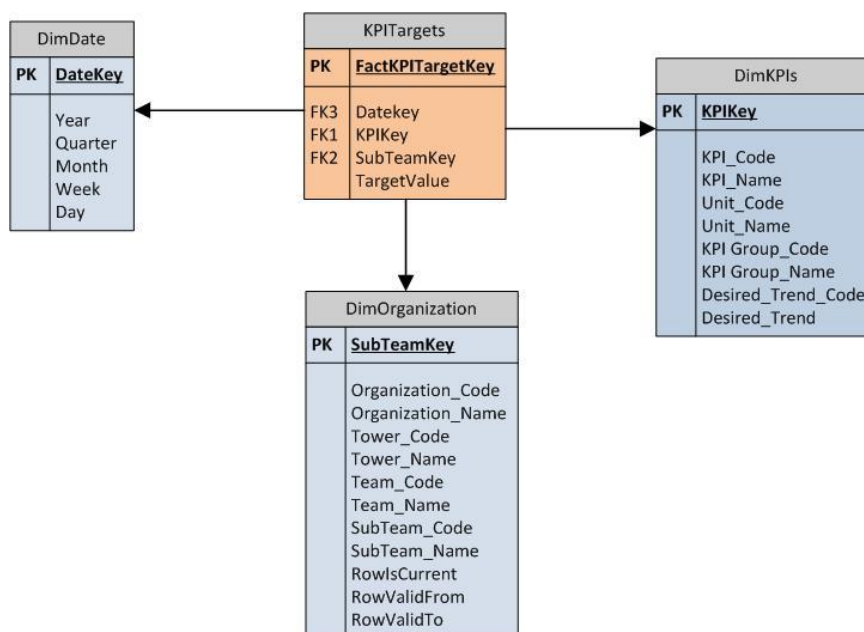


Figura 4.7: Schema logico del *data mart* KPITargets

4.11 Progettazione logica del *data warehouse*

È molto semplice vedere come i tre *data mart* abbiano in comune diverse dimensioni. Pertanto si ottiene uno schema a costellazione, rappresentato in Figura 4.8.

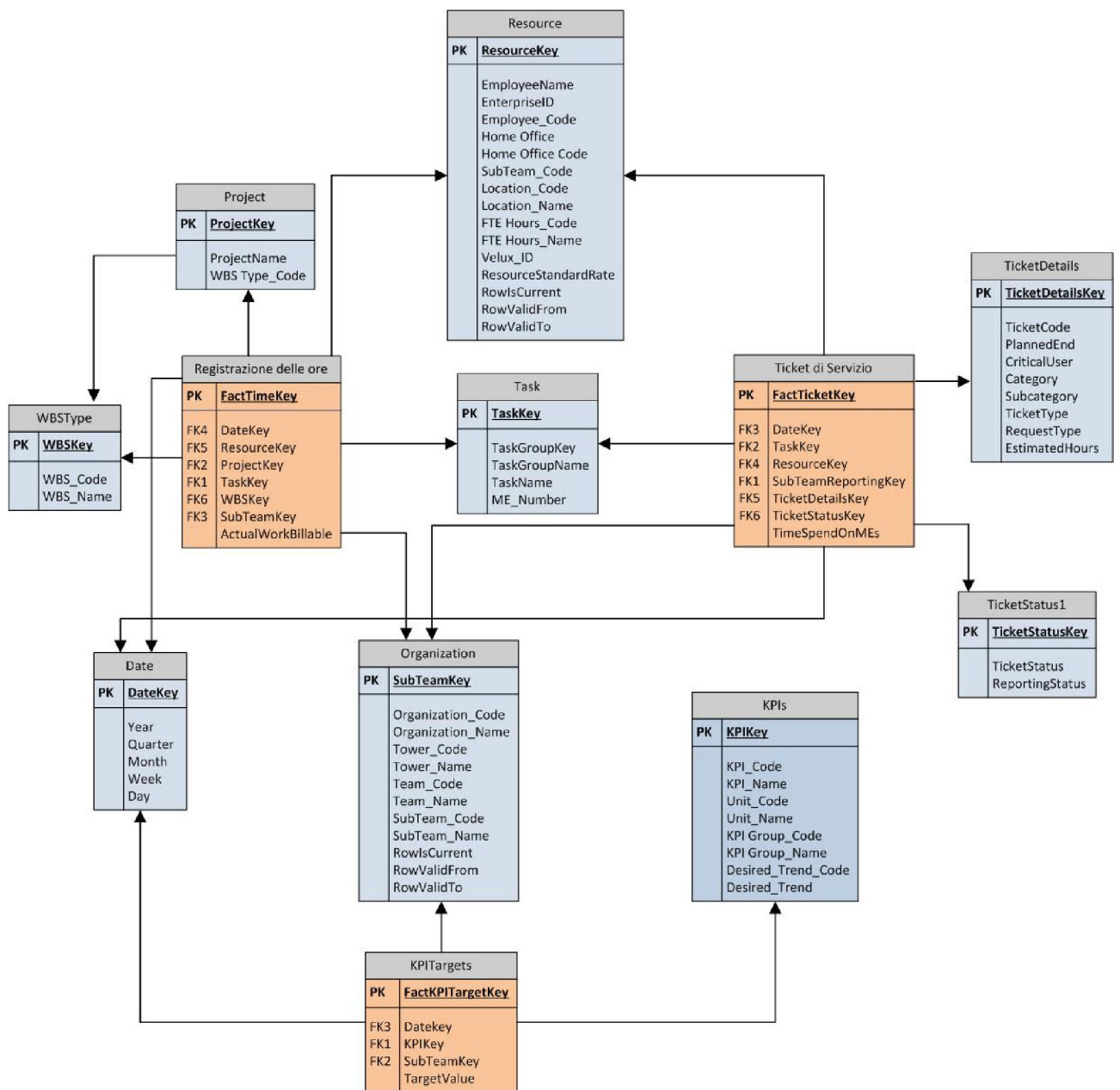


Figura 4.8: Schema a costellazione del *data warehouse*

AMBIENTE E STRUMENTI DI SVILUPPO

In questo capitolo viene descritto il DataBase Management System utilizzato per sviluppare il progetto. Sono illustrate le caratteristiche generali di Microsoft SQL Server 2012 Enterprise e di tutte le sue componenti utilizzate. Durante la presentazione delle componenti vengono anche elencate le scelte realizzative effettuate, eccezion fatta per lo strumento di ETL e per quello di Reportistica, ai quali sono dedicati i due capitoli successivi di questo documento.

5.1 Considerazioni sulla scelta dell'ambiente Microsoft

Sul mercato è disponibile una vasta gamma di software di Business Intelligence, ognuno dei quali mira ad offrire il miglior compromesso in termini di qualità, prestazioni, usabilità e prezzo per analizzare la grande mole di dati aziendali, al fine di riuscire a migliorare le scelte decisionali. La tecnologia Microsoft è in grado di offrire una soluzione completa, sicura e affidabile di Business Intelligence grazie ai suoi numerosi componenti che permettono di raccogliere, analizzare, trasformare, integrare e misurare dati. La realizzazione della piattaforma di Business Intelligence e la presentazione delle analisi multidimensionali sono fortemente integrate a seguito dell'utilizzo della medesima tecnologia, favorendo una migliore collaborazione, comunicazione e presentazione dei dati. Grazie a numerose estensioni fornite a completamento del pacchetto, i dipendenti aziendali possono godere del vantaggio di utilizzare tecnologie già note, quali Microsoft

Office ed Excel, per poter esplorare i dati, con una significativa riduzione dei tempi di apprendimento. La D60 vanta, fin dalla sua fondazione, una consolidata partnership con Microsoft, i cui prodotti, nel corso degli anni, sono sempre risultati in linea con le esigenze espresse dall'azienda.

5.2 SQL Server 2012 Enterprise

SQL Server 2012 è il DBMS (DataBase Management System) utilizzato per la realizzazione del processo di data warehousing. Nella versione Enterprise, viene proposta una piattaforma integrata per la gestione dei dati e della Business Intelligence, in grado di fornire scalabilità, strumenti di integrazione dati (ETL), applicazioni di data warehousing, sicurezza, gestione dei dati, analisi e reportistica avanzata. Rispetto alle precedenti versioni del prodotto, SQL Server 2012 introduce numerosi componenti e funzionalità aggiuntive, tra le principali è doveroso ricordare:

- **xVelocity.** Il motore di analisi in memoria xVelocity, evoluzione del motore VertiPaq introdotto in SQL Server 2008 R2 con PowerPivot per Excel 2010 e PowerPivot per SharePoint 2010, è un motore *columnstore* che raggiunge prestazioni eccellenti nelle query analitiche mediante tecniche quali l'archiviazione a colonne, la compressione avanzata, la memorizzazione nella cache in memoria e algoritmi di analisi dei dati e aggregazione altamente paralleli;
- **PowerView.** Un pannello web per la presentazione e visualizzazione dei dati;
- Nuovi strumenti per la gestione e integrazione dei dati fra cui **SQL Server Data Quality Services** e **Master Data Services**;
- **Enhanced PowerPivot.** Uno strumento per dare nuove possibilità di analisi dei dati ed una più facile integrazione con Excel;
- **Tabular Model.** Un nuovo modello per la Business Intelligence;
- **SQL Server Data Tools.** Un nuovo strumento che unifica lo sviluppo su SQL Server e SQL Azure, la versione cloud del DBMS.

5.2.1 Management studio

SQL Server Management Studio è un ambiente integrato per l'accesso, la configurazione, la gestione e l'amministrazione di tutti i componenti di SQL Server. Come evidenziato da [Msdn 13], tramite un gruppo esteso di strumenti grafici ed un editor di script (tra i tanti, TSQL, MDX e XML) è possibile creare, gestire, amministrare e programmare una qualsiasi base di dati.

Nel corso del progetto è stato utilizzato innumerevoli volte per la consultazione dei dati, il testing e la valutazione dell'efficacia di query SQL e T-SQL poi utilizzate all'interno dei componenti di Integration Services durante la fase di ETL.

5.2.2 Integration Services

Microsoft Integration Services è una piattaforma per la compilazione di soluzioni di integrazione e trasformazione di dati a livello aziendale grazie alla quale, come ribadito da [Msdn 13], è possibile risolvere problemi aziendali complessi, tramite operazioni di copia o download di file, invio di messaggi di posta elettronica in risposta a determinati eventi, aggiornamento di *data warehouse*, pulizia dei dati, data mining e gestione di oggetti e dati di SQL Server. In Integration Services è possibile estrarre e trasformare i dati da un'ampia varietà di origini quali file di dati XML, file flat e origini dati relazionali, quindi caricare i dati in una o più destinazioni.

“È possibile utilizzare gli strumenti grafici di Integration Services per creare soluzioni senza scrivere una sola riga di codice oppure programmare il modello a oggetti completo di Integration Services per creare pacchetti a livello di codice e codificare attività personalizzate e altri oggetti di pacchetti” [Msdn 13]. Nel quinto capitolo di questo documento verrà dettagliata la fase di estrazione, trasformazione e caricamento dati e verrà specificato quali componenti di Integration Services sono stati utilizzati per risolvere al meglio le problematiche riscontrate.

5.2.3 Analysis Services

Analysis Services è il motore OLAP di SQL Server. Dotato di tecnologie che semplificano e velocizzano il processo di progettazione, creazione, gestione e interrogazione delle strutture multidimensionali, fornisce un'origine dati OLAP agli strumenti di creazione e modifica di business intelligence come Microsoft Excel, PerformancePoint Dashboard Designer e Visio. Come riportato

da [Msdn 13], alla base di qualsiasi soluzione di Analysis Services c'è un modello di dati Business Intelligence e un'istanza del server. I modelli utilizzano dati storici che sono già raccolti in database transazionali e altri archivi dati, quindi vengono annotati con metadati che consentono di misurare, modificare e confrontare i dati aziendali in query ad hoc o report personalizzati. I modelli vengono poi distribuiti a un server Analysis Services come database, dove diventano disponibili a utenti autorizzati che si connettono tramite Excel e altri strumenti.

La versione di Analysis Services distribuita con SQL Server 2012 aggiunge un importante nuovo approccio di modellazione tabulare che risulta essere “molto intuitiva se si è un business analyst abituato all'utilizzo di dati relazionali” [Msdn 13]. La modellazione tabulare è diversa perché si compila un modello semantico Business Intelligence mediante tabelle e relazioni piuttosto che mediante cubi e dimensioni.

Per lo sviluppo del progetto sono state implementate entrambe le soluzioni. È stato sviluppato in principio il modello multidimensionale e in seguito quello tabulare. Le due successive sottosezioni sono dedicate alla descrizione delle caratteristiche salienti di entrambi i modelli che vengono infine messi a confronto, spiegando le ragioni che hanno portato alla decisione di implementarli entrambi.

5.2.3.1 Modellazione Multidimensionale (SSAS)

Una soluzione multidimensionale Analysis Services utilizza strutture multidimensionali per l'analisi dei dati aziendali in più dimensioni. La modalità multidimensionale è la modalità server predefinita di Analysis Services, in cui, come dettagliato da [Msdn 13], è incluso un motore di calcolo per i dati OLAP, con modalità di archiviazione MOLAP, ROLAP e HOLAP per bilanciare prestazioni con requisiti di scalabilità. Il motore OLAP di Analysis Services è leader del settore e offre ottimi risultati con un'ampia gamma di strumenti di Business Intelligence.

Il motivo primario per la compilazione di un modello multidimensionale di Analysis Services è la realizzazione di prestazioni veloci delle query a hoc sui dati aziendali. Un modello multidimensionale è costituito da cubi e dimensioni che possono essere annotati ed estesi per supportare costruzioni di query complesse. “Data la crescente importanza della Business Intelligence a tutti i livelli di un'organizzazione, la disponibilità di una singola origine di dati analitici assicura che le discrepanze siano limitate al minimo, se non eliminate completamente” [Msdn 13].

Un altro importante vantaggio derivante dall'utilizzo di database multidimensionali di Analysis Services, evidenziato da [Msdn 13], è l'integrazione con gli strumenti di generazione report di Business Intelligence di uso comune quali Excel, Reporting Services e PerformancePoint, nonché di applicazioni personalizzate e soluzioni di terze parti.

I cubi sono gli oggetti query fondamentali in Analysis Services. Quando si esegue una connessione a un database di Analysis Services tramite un'applicazione client, si esegue in pratica la connessione a un cubo all'interno di tale database.

Analysis Services è stato utilizzato per la creazione del cubo, per la definizione delle gerarchie tra i dati e per il calcolo, tramite il linguaggio MDX (Multidimensional Expressions), degli indicatori di performance in linea con la progettazione descritta nel terzo e nel quarto capitolo di questo documento. In fase di testing, tuttavia, è stato riscontrato nel modello un elevato tempo di risposta rispetto alle aspettative. Questo aspetto, di cruciale importanza in un progetto di data warehousing, ha portato a riconsiderare le procedure di ETL in modo tale da effettuare alcuni pesanti calcoli in quella fase del processo, alleggerendo così le operazioni di aggregazione effettuate sul cubo al fine di migliorare le prestazioni. Questo aspetto verrà approfondito nel prossimo capitolo. Un'alternativa a questa soluzione è rappresentata dallo sviluppo di un secondo modello, quello tabulare.

5.2.3.2 Modellazione Tabulare (SSAS Tabulare)

I modelli tabulari sono database in memoria in Analysis Services. Utilizzando un processore di query multithreading e algoritmi di compressione all'avanguardia, il motore di analisi in memoria xVelocity (VertiPaq), le cui caratteristiche sono ampiamente descritte da, offre delle ottime performance in termini di accesso ai dati.

I modelli tabulari supportano l'accesso ai dati tramite due modalità, ovvero cache e DirectQuery. “In modalità cache è possibile integrare i dati da più origini, tra cui database relazionali, feed di dati e file di testo. In modalità DirectQuery è possibile ignorare il modello in memoria, consentendo l'esecuzione di una query sui dati direttamente nell'origine (database relazionale di SQL Server) da parte delle applicazioni client” [Msdn 13].

I modelli tabulari vengono creati in SQL Server Data Tools (SSDT) utilizzando tabelle e relazioni come strutture primarie per la rappresentazione di dati, i quali vengono importati da più origini, con la possibilità di affinare il modello aggiungendo relazioni, colonne calcolate, misure, indicatori di performance e gerarchie. I modelli possono quindi essere distribuiti in un'istanza di Analysis

Services dove, tramite applicazioni client di creazione di report, è possibile connettersi a loro. Tali modelli, inoltre, possono essere partizionati per elaborazioni ottimizzate e protetti a livello di riga tramite ruoli basati sulla sicurezza.

Per via delle loro caratteristiche, come ribadito da [Msdn 13], i database modello tabulare sono più facili da gestire rispetto ai database multidimensionali, specialmente se si dispone di un background nell'ambito dei database relazionali, mentre le attività amministrative sono molto simili a quelle di un database multidimensionale. Nella Tabella 5.1 vengono messe a confronto, in termini di vantaggi e svantaggi, la modellazione tabulare e la modellazione multidimensionale.

	Modellazione Tabulare	Modellazione Multidimensionale
Vantaggi	<ul style="list-style-type: none"> • Offre migliori performance rispetto a OLAP nella maggioranza dei casi; • Implementazione più semplice e intuitiva; • Integrazione con Power Pivot. 	<ul style="list-style-type: none"> • Tecnologia solida; • Capacità di gestire un'elevata mole di dati;
Svantaggi	<ul style="list-style-type: none"> • Tecnologia ancora in fase di sviluppo; • Alcune operazioni complesse implementabili MDX non sono disponibili nel linguaggio del modello tabulare, il DAX. 	<ul style="list-style-type: none"> • A seconda del contest, prestazioni poco performanti; • Non sono previste significative innovazioni nell'immediato futuro; • Complessità più elevata rispetto al modello tabulare.

Tabella 5.1: Confronto tra la modellazione multidimensionale e la modellazione tabulare.

Il linguaggio utilizzato nella modellazione tabulare è il DAX (Data Analysis Expressions). DAX è un linguaggio che consente agli utenti di definire calcoli personalizzati nelle tabelle di PowerPivot (colonne calcolate) e nelle tabelle pivot di Excel (misure). In DAX sono incluse alcune delle funzioni utilizzate nelle formule di Excel e funzioni aggiuntive progettate per utilizzare dati relazionali ed eseguire aggregazioni dinamiche, ragion per cui risulta essere molto intuitivo per gli

utenti che hanno una certa familiarità con Excel. Per creare una formula, si digita un segno di uguale, seguito da un nome di funzione o da un'espressione e dai valori o dagli argomenti necessari. Analogamente a Excel, DAX offre numerose funzioni che è possibile utilizzare per lavorare con le stringhe, eseguire calcoli utilizzando date e ore o creare valori condizionali.

Il modello tabulare realizzato, sebbene adotti una logica totalmente differente rispetto a quello multidimensionale, permette di ottenere risultati totalmente identici (in termini di indicatori di performance), con tempi di risposta molto più rapidi. Tuttavia, a causa del problema di interfacciamento con il server HPSC affrontato nel quarto capitolo di questo documento, non è stato possibile effettuare il test sulla totalità dei dati. Pertanto, sebbene entrambe le soluzioni siano state implementate, non è stato stabilito, entro il termine del tirocinio, quale delle due tecnologie utilizzare, rimandando la decisione al momento in cui sarà possibile effettuare i test completi.

5.2.4 Reporting Services

“SQL Reporting Services è una piattaforma di reporting basata su server che fornisce funzionalità di report complete per numerose origini dati, nonché una gamma completa di strumenti e servizi pronti all'uso che consentono di creare, distribuire e gestire report” [Msdn 13]. Progettati tramite Microsoft Visual Studio o tramite particolari tool come ad esempio Report Builder (i quali tuttavia offrono una gamma limitata degli strumenti e delle potenzialità offerte da Visual Studio), i report vengono definiti con il linguaggio RDL (Report Definition Language) e possono essere esportati in numerosi formati (PDF, CSV, XML, Excel ecc.). Gli utenti possono visualizzare i report tramite un'interfaccia web e l'accessibilità è regolata da un solido sistema role-based. L'ultimo capitolo di questo documento è dedicato alla reportistica, ragion per cui la trattazione di SSRS verrà approfondita in seguito. Verranno mostrati alcuni dei report prodotti tramite Reporting Services e verranno messi a confronto con i corrispettivi report iniziali che si desiderava riprodurre.

5.2.5 Master Data Services

Master Data Services (MDS) è la soluzione SQL Server per la gestione dei dati master, definiti come le informazioni critiche per il corretto funzionamento dei processi. I dati master, che comprendono generalmente informazioni su clienti, prodotti, personale, materiali, fornitori e simili, sono in prevalenza di natura non transazionale e molto spesso sono immagazzinati in diverse origini dati, ragion per cui si corre spesso il rischio di incorrere in duplicazioni e inconsistenze. Master Data Services è gestore dei dati master (Master Data Management) della Microsoft, appositamente progettato per fornire una sorgente di dati centralizzata e mantenere i dati master sincronizzati in modo tale da evitare ridondanze e inconsistenze.

In un progetto di gestione dei dati master in genere è inclusa la valutazione e la ristrutturazione dei processi aziendali interni insieme all'implementazione della tecnologia di gestione dei dati. Il risultato che si vuole ottenere è rappresentato da dati affidabili e centralizzati che possono essere analizzati e dar luogo a decisioni aziendali ottimali.

Tra le varie funzionalità, Master Data Services offre gestione delle gerarchie, sicurezza granulare, controllo delle versioni dei dati e definizioni di regole business. In sintesi, come riporta ampiamente [Msdn 13], Master Data Services consente di:

- **creare e gestire dati master definendo uno standard per le informazioni chiave.** Offre strumenti specializzati per la raccolta e il mantenimento dell'accuratezza dei dati master, permettendo di gestirne la consistenza qualunque sia la loro origine, inclusi ambienti quali Oracle, Microsoft SQL Azure, HP, e IBM. Consente di creare e gestire gerarchie e di preservare quelle preesistenti nel caso in cui vengano rilevate al momento dell'importazione dei dati;
- **creare una sorgente centralizzata di dati per l'impresa.** La creazione di una sorgente dati centralizzata consente di mantenere un certo livello di consistenza e sincronizzazione evitando che sorgano anomalie tra le varie applicazioni che fanno uso dei dati. È possibile tenere traccia delle modifiche apportate in diversi intervalli di tempo, definire ruoli e business rules;

L'interfaccia utente o back-end è intuitiva e di semplice utilizzo, strutturata in modo tale da ridurre al minimo i possibili errori dovuti all'inserimento manuale dei dati, qualunque sia il livello di esperienza nel settore del personale che la utilizza.

La Figura 5.2 mostra uno schema riassuntivo del flusso di dati tramite Master Data Services.

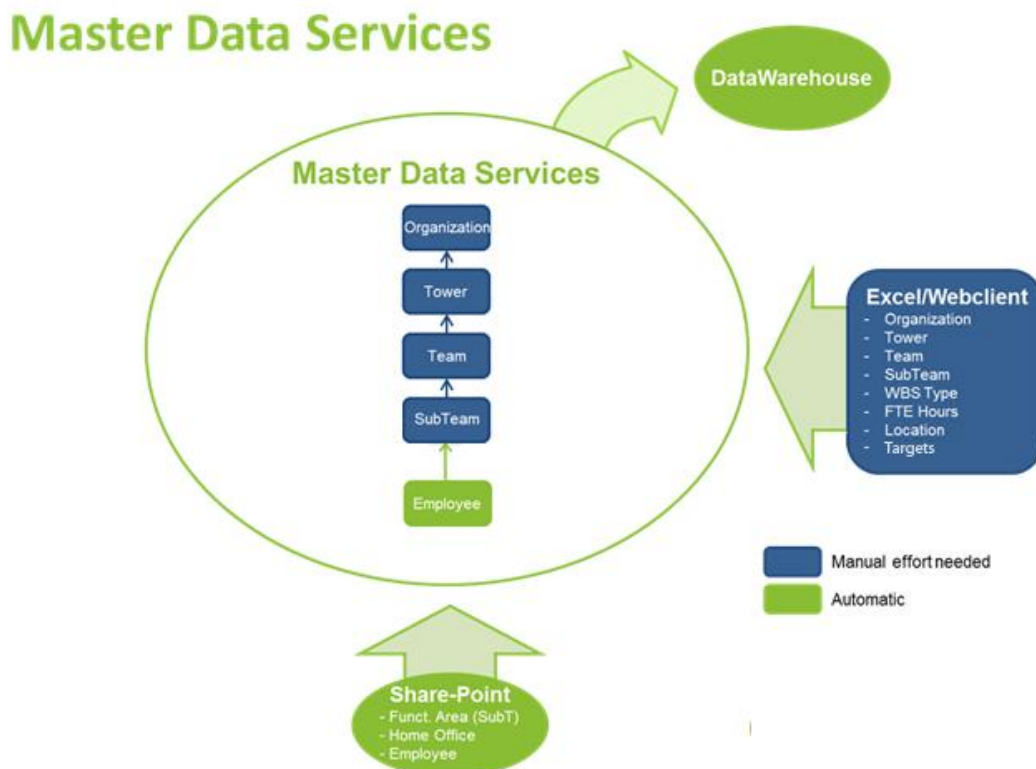


Figura 5.2: Schema riassuntivo del flusso di dati tramite Master Data Services.

Come accennato nel capitolo precedente (Sezione 4.4), Master Data Services ha svolto un ruolo fondamentale per la corretta risoluzione del progetto. Come si evince dalla Figura 5.2, i dati sui dipendenti estratti da SharePoint, la gerarchia organizzativa, le tipologie WBSType e i valori target degli indicatori di performance vengono validati, completati e aggiornati dai responsabili Accenture tramite MDS prima di popolare le dimensioni. Sebbene tutto ciò richieda uno sforzo manuale da parte del personale, lo sforzo più impegnativo per effettuare il cosiddetto “mapping” deve essere effettuato solo la prima volta. D'altra parte, per costruire un modello conforme alle richieste del committente è necessario che parte del personale interno provveda a colmare le lacune riscontrate nei dati.

È una soluzione centralizzata che permette di visualizzare, mappare e validare i dati ed è strutturata in modo tale da ridurre al minimo le possibili inconsistenze ed errori. La manutenzione dei dati viene effettuata sfruttando il web client di MDS. Accedendo alla sezione “Explorer”, è

possibile visualizzare l'elenco delle cosiddette “Entities”, ovvero la lista delle tabelle operazionali sulle quali è possibile operare.



Figura 5.3: Elenco parziale delle tabelle operazionali sulle quali è possibile operare tramite MDS.

Una volta definite le associazioni tra le tabelle, l'inserimento di nuovi dati o la modifica di quelli esistenti diventa una procedura guidata. Ad esempio, se si volesse modificare un progetto esistente, la scelta della tipologia è delimitata a un insieme di valori ricavati dalla tabella di riferimento. Lo stesso concetto vale per tutte le altre tabelle rese disponibili in MDS e ciò permette di preservare una certa consistenza tra i dati e limitare al minimo lo sviluppo di errori dovuto all'inserimento manuale.

Figura 5.4: Elenco parziale delle tabelle operazionali sulle quali è possibile operare tramite MDS.

Una funzionalità interessante di MDS è la cosiddetta “Transaction List”. La Transaction List (mostrata in Figura 5.5) consiste in un elenco delle modifiche apportate tramite il quale è possibile visualizzare le annotazioni ad esse associate (se inserite al momento della modifica come mostra la parte finale della Figura 5.4). Consultandolo, le persone incaricate hanno la possibilità di visualizzare le ragioni per le quali sono state apportate determinate modifiche in modo da mantenere una certa coerenza ed evitare errori dovuti a incomprensioni tra il personale come, a detta del committente, avveniva in passato.

View Transactions for 2094 {Base - Map and Guide - After Sales Systems Group_May_Oct_2012}

Attribute	Hierarchy	Date ▼	Prior Value	New Value	User	Transaction
WBS Type		10/23/2012 11:54:30 AM		1	DIR\klaus.g.ulsoe	Attribute value specified
SubTeam		10/23/2012 11:54:30 AM		18	DIR\klaus.g.ulsoe	Attribute value specified
WBS Type		10/23/2012 10:55:31 AM	1		DIR\klaus.g.ulsoe	Attribute value specified
SubTeam		10/23/2012 10:55:31 AM	46		DIR\klaus.g.ulsoe	Attribute value specified
SubTeam		10/23/2012 10:54:48 AM	45	46	DIR\klaus.g.ulsoe	Attribute value specified
WBS Type		10/23/2012 10:53:59 AM		1	DIR\klaus.g.ulsoe	Attribute value specified
SubTeam		10/23/2012 10:53:59 AM		45	DIR\klaus.g.ulsoe	Attribute value specified

1-10 of 10 transactions

Annotations

DIR\klaus.g.ulsoe 10/23/2012 11:54:30 AM

some notes

Add Annotation

Close

Figura 5.5: Transactions List.

È ovviamente possibile stabilire delle regole sui permessi di accesso, in modo tale che ogni dipendente possa visualizzare e modificare solo le tabelle di sua competenza. Valutazioni analoghe valgono per il settaggio e l'aggiornamento dei valori target degli indicatori di performance. Una volta validati tramite MDS, i dati vengono raccolti e utilizzati per aggiornare il *data warehouse*.

PROCEDURE DI ESTRAZIONE, TRASFORMAZIONE E CARICAMENTO

Si descrive il processo di estrazione, trasformazione e caricamento (ETL) dei dati al fine di predisporli correttamente al caricamento nel *data warehouse*, in linea con le analisi precedentemente descritte. Si presenta il software utilizzato per l'implementazione e lo sviluppo delle procedure di ETL, descrivendo le sue principali componenti con particolare attenzione a quelle maggiormente utilizzate. Al contempo si illustrano i motivi di alcune scelte architetturali, si presentano le principali problematiche riscontrate durante lo sviluppo delle procedure ETL e per ciascuna di esse viene descritto il modo in cui è stata affrontata. Per concludere il capitolo, vengono presentati alcuni schemi riassuntivi del flusso di dati.

6.1 Il processo di estrazione, trasformazione e caricamento dati

Una volta progettato il *data warehouse* si passa alla sua realizzazione e si affronta l'importante problema dell'estrazione, trasformazione e caricamento dei dati (Extract, Transform, Load). Come ricordano [Albano 12] e [Ruggieri 12], i dati da inserire nel *data warehouse* vanno prima sottoposti a due importanti e complesse operazioni:

- **Trasformazione.** Quando i dati provengono da basi di dati diverse, i loro formati vanno rivisti per renderli omogenei eliminando differenze sintattiche e semantiche. Gli stessi dati

possono avere sia attributi con nomi diversi nelle basi di dati di provenienza, e i nomi non sono quelli che si vogliono usare nel *data warehouse*, sia tipi diversi. Per esempio, un codice definito in alcuni casi di tipo stringa e in altri di tipo intero, il sesso definito come (M, F), (m, f), (0, 1) o (maschio, femmina) oppure un valore definito con unità di misura diversa. In questo caso si parla di trasformazioni sintattiche. È possibile che i dati nelle basi di dati di provenienza possano essere stati usati con un significato diverso. Per esempio, le vendite possono essere giornaliere oppure settimanali. In questo caso si dovrebbero operare le cosiddette trasformazioni semantiche.

- **Pulizia.** I dati vanno analizzati per eliminare errori di rappresentazione o per completarli con informazioni mancanti. Per esempio, nel caso di indirizzi può essere errato il CAP o il nome del Comune.

Un sistema ETL ben disegnato estrae i dati da sorgenti separate, li controlla e assicura che i dati siano conformi a determinati standard di qualità e di consistenza stabiliti al momento della progettazione del *data warehouse*, rende omogenei i dati provenienti dalle diverse sorgenti e infine li inserisce nella base di dati di destinazione. L'efficienza del funzionamento di un *data warehouse* è influenzata dal modo con cui vengono progettate e realizzate le procedure di ETL.

L'implementazione delle procedure di ETL è stata effettuata subito dopo l'analisi concettuale e logica del *data warehouse* e alcune fasi sono state rivisitate in seguito a causa delle problematiche legate alle performance delle quali si è discusso nel precedente capitolo. Per realizzare le procedure di ETL ci si è avvalsi di un potente strumento già consolidato sul mercato: SQL Server Integration Services 2012 (SSIS).

6.2 Vantaggi dell'utilizzo di SQL Server Integration Services (SSIS)

Esistono due alternative nella scelta dello strumento di cui servirsi per lo sviluppo delle procedure di ETL: si può utilizzare uno strumento disponibile in commercio oppure si possono implementare le procedure manualmente, avvalendosi delle librerie a disposizione degli sviluppatori. Per confrontare le due alternative ci serviremo di una tabella simile a quella utilizzata nella Sottosezione 5.2.3.2. Nella Tabella 6.1 vengono messe a confronto, in termini di vantaggi e svantaggi, la scelta di utilizzare un software in commercio e la scelta di implementare le procedure manualmente.

	Software in commercio	Implementazione manuale
Vantaggi	<ul style="list-style-type: none"> • Semplicità di utilizzo; • Riduzione dei tempi di sviluppo; • Possibilità di utilizzare i componenti innumerevoli volte; • Affidabilità. 	<ul style="list-style-type: none"> • Flessibilità potenzialmente illimitata; • Controllo ad hoc sulle trasformazioni;
Svantaggi	<ul style="list-style-type: none"> • Costi; • Minore flessibilità. 	<ul style="list-style-type: none"> • Complessità (richiede conoscenze avanzate del linguaggio di programmazione senza le quali è impossibile sia implementare le procedure che contribuire alla manutenzione del sistema); • Allungamento dei tempi di sviluppo (implementazione, testing).

Tabella 6.1: Confronto tra la scelta di utilizzare un software in commercio e l'implementazione manuale.

Dalla Tabella 6.1 si evince che a meno di casi estremi quali la presenza di un budget estremamente limitato o la presenza di operazioni di trasformazione particolarmente complesse, conviene affidarsi a un software in commercio. Come detto nel capitolo precedente, la D60 vanta, fin dalla sua fondazione, una consolidata partnership con Microsoft. La politica aziendale prevede che il personale addetto alla manutenzione del sistema possa differire da quello addetto allo sviluppo del sistema stesso, pertanto è auspicabile che le procedure utilizzate per l'implementazione siano intuitive e di facile comprensione in modo da agevolare le operazioni di manutenzione. Infine, il caso specifico non ricade in nessuno di quelli appena elencati tra i casi estremi. Per tutte queste

ragioni è stato scelto di utilizzare il modulo proprietario Integration Services per lo sviluppo delle procedure di ETL.

6.3 Struttura del pacchetto SSIS

Un pacchetto SSIS (o package) è una collezione organizzata di connessioni, elementi di flusso di controllo, elementi di flusso di dati, gestori eventi, variabili, parametri e configurazioni che possono essere assemblati o compilati a livello di codice o utilizzando gli strumenti di progettazione grafica disponibili in SQL Server Integration Services. Al momento della creazione un pacchetto è un oggetto vuoto, privo di funzionalità. Per aggiungere funzionalità a un pacchetto è necessario aggiungervi un flusso di controllo e, facoltativamente, uno o più flussi di dati [Msdn 13].

Nella Figura 6.2 viene illustrato un semplice pacchetto che contiene un flusso di controllo con un task del flusso di dati, che a sua volta contiene un flusso di dati.

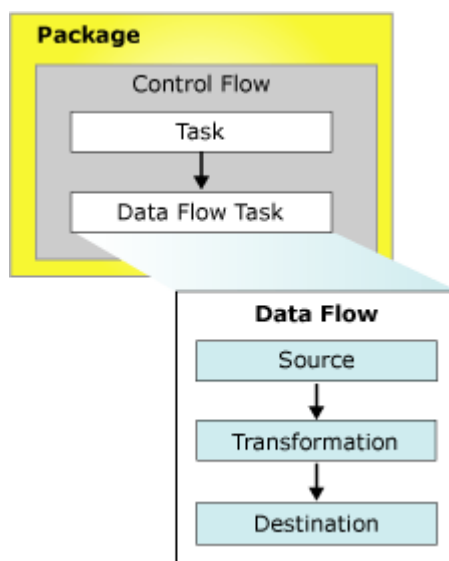


Figura 6.2: Pacchetto che contiene un flusso di controllo con un task del flusso di dati [Msdn 13].

6.3.1 Flusso di controllo

Un pacchetto è costituito da un flusso di controllo e, facoltativamente, da uno o più flussi di dati. In SQL Server Integration Services sono disponibili tre diversi tipi di elementi del flusso di controllo:

contenitori, che definiscono le strutture nei pacchetti, task, che forniscono le funzionalità, e vincoli di precedenza, che connettono eseguibili, contenitori e task in un flusso di controllo ordinato.

Nella Figura 6.3 viene illustrato un flusso di controllo con un contenitore e sei task, di cui cinque definiti a livello di pacchetto e uno a livello di contenitore. Tale task è all'interno di un contenitore.

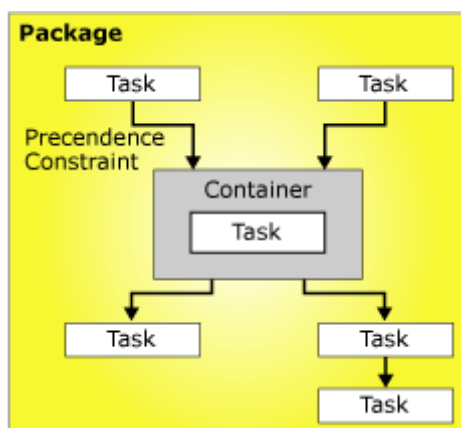


Figura 6.3: Flusso di controllo [Msdn 13].

Per controllare l'ordine o definire le condizioni per l'esecuzione del task o del contenitore successivo nel flusso di controllo del pacchetto, è necessario connettere i task e i contenitori del pacchetto tramite vincoli di precedenza. Tali vincoli possono essere corredati da condizioni su variabili del pacchetto, che permettono la scelta di un ramo piuttosto che di altri, e uniti secondo operatori AND/OR, in modo da dare la possibilità di seguire logiche condizionate e più complesse. È anche possibile raggruppare sottoinsiemi di task e contenitori per eseguirli ripetutamente come unità nell'ambito del flusso di controllo del pacchetto. Vengono ora descritti nel dettaglio alcuni tra i componenti principali del flusso di controllo SSIS.

- **Data Flow Task.** Incapsula i task del flusso di dati che muovono i dati dalle sorgenti alle destinazioni provvedendo alla trasformazione, pulitura e modifica dei dati trattati.

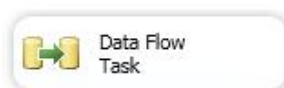


Figura 6.4: Data Flow Task.

- **Execute SQL Task.** Esegue una query SQL personalizzata dall'utente.

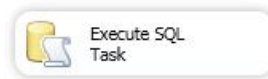


Figura 6.5: Execute SQL Task.

- **Sequence Container.** Contenitore all'interno del quale è possibile definire un ordine di esecuzione di task del flusso di dati e SQL task.

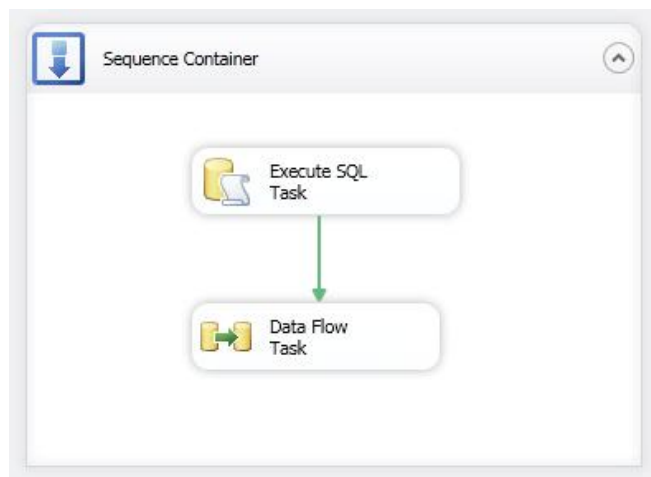


Figura 6.6: Sequence Container.

- **Execute Package.** Abilita l'esecuzione di un package; se collegati, questi task eseguono sequenzialmente i vari package che richiamano.

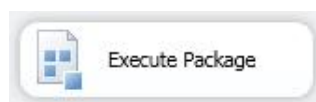


Figura 6.7: Execute Package.

6.3.2 Flusso di dati

In SQL Server Integration Services sono disponibili tre diversi tipi di componenti flusso di dati: origini, trasformazioni e destinazioni. Le origini estraggono dati da archivi dati quali tabelle e viste di database relazionali, file e database di Analysis Services. Tramite le trasformazioni è possibile modificare, riepilogare e pulire i dati. Le destinazioni consentono di caricare dati in archivi dati o di creare set di dati in memoria. Per connettere i componenti flusso di dati è necessario connettere l'output di origine all'input di trasformazioni e destinazioni [Msdn 13]. Nella Figura 6.8 viene illustrato un flusso di dati che include un'origine, una trasformazione con un input e un output e una destinazione. Oltre alle colonne di input, di output ed esterne, la Figura 6.8 include anche gli input, gli output e gli output degli errori.

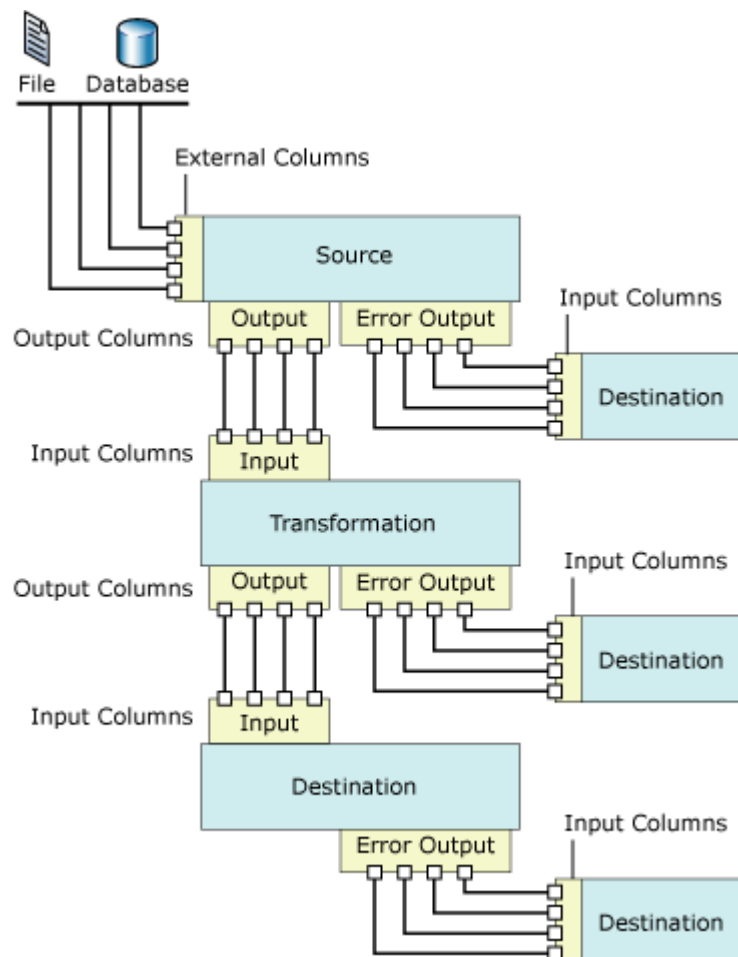


Figura 6.8: Flusso di dati [Msdn 13].

Le origini e le destinazioni devono essere associate a un Connection Manager. I connettori disponibili riguardano svariate tipologie di risorse esterne, come ODBC, OLEDB, EXCEL, FLAT File, FTP, SMTP, File, e via discorrendo. Vengono ora descritti nel dettaglio i componenti principali del flusso di dati SSIS utilizzati nel progetto, divisi per categoria.

- **(Origine) OLE DB Source.** Estrae i dati da un data base relazionale usando un provider OLE DB.



Figura 6.9: OLE DB Source.

- **(Origine) Flat File Source.** Estrae i dati da un file statico.



Figura 6.10: Flat File Source.

- **(Trasformazione) Data Conversion.** Trasformazione che converte il tipo di dati di una colonna in un tipo di dati diverso.

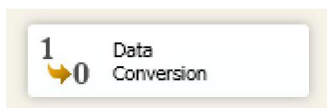


Figura 6.11: Data Conversion.

- **(Trasformazione) Conditional Split.** Direzione le righe di un data set sulla base di condizioni definite dall'utente.

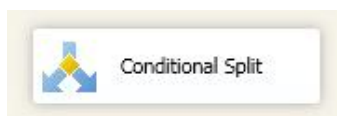


Figura 6.12: Conditional Split.

- **(Trasformazione) Multicast.** Trasformazione che distribuisce set di dati tra più output.



Figura 6.13: Multicast.

- **(Trasformazione) Derived Column.** Trasformazione che popola colonne con risultati di espressioni.



Figura 6.14: Derived Column.

- **(Trasformazione) Union All.** Trasformazione che unisce due set di dati ordinati.



Figura 6.15: Union All

- **(Trasformazione) OLE DB Command.** Trasformazione che esegue comandi SQL per ogni riga di un flusso di dati.

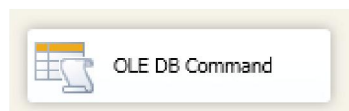


Figura 6.16: OLE DB Command.

- **(Trasformazione) Script Component.** Presente in tre tipologie: Source, funge da origine nel flusso di dati e restituisce dati in output; Destination, funge da destinazione nel flusso di dati e riceve dati in ingresso; Transformation, funge da trasformazione nel flusso di dati, riceve dati in ingresso e restituisce dati in uscita). È una trasformazione che utilizza uno

script (il cui linguaggio può essere, opzionalmente, C# o Visual Basic) per estrarre, trasformare o caricare dati.

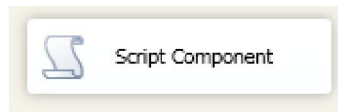


Figura 6.17: Script Component.

- **(Trasformazione) Lookup.** La trasformazione Lookup consente di eseguire ricerche unendo in join i dati contenuti nelle colonne di input con le colonne in un set di dati di riferimento.



Figura 6.18: Lookup.

- **(Destinazione) OLE DB Destination.** Carica i dati all'interno di un data base relazionale utilizzando un provider OLE DB.

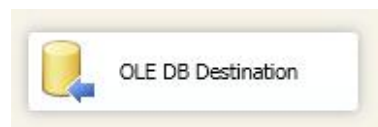


Figura 6.19: OLE DB Destination.

6.4 Nomenclatura

Dal momento che, come accennato, la politica aziendale prevede che il personale addetto alla manutenzione del sistema possa differire da quello addetto allo sviluppo del sistema stesso, la definizione di uno standard diventa un aspetto non trascurabile e permette di agevolare il compito del personale addetto alla manutenzione. Di seguito vengono specificate le regole utilizzate per la nomenclatura dei componenti del flusso di controllo e del flusso di dati.

- SEQ_Q: Sequence Containter;
- SQL: Query SQL;
- DFT: Data Flow Task;
- OLE_SRC: OLEDB Source;
- OLE_DST: OLEDB Destination;
- FFS: Flat File Source;
- LKP: LookUp;
- DCVN: Data Conversion;
- DER: Derived Column;
- CMD: Command;
- CSPL: Conditional Split;
- SCR: Script Component.

Sono stati definiti degli standard anche per la nomenclatura delle tabelle. Le tabelle importate nella staging area mantengono il loro nome originale con l'aggiunta di un prefisso indicante la sorgente dati dalla quale provengono (MSP, HPSC e MDS), le tabelle che gestiscono il cambiamento storico sono caratterizzate dal prefisso SCD, mentre quelle relative al *data warehouse* dal prefisso DW.

6.5 Staging Area

Come anticipato nei precedenti capitoli, l'architettura utilizzata per la realizzazione dell'intero processo di data warehousing prevede l'utilizzo di un livello intermedio, tra le sorgenti e il *data warehouse*, chiamato staging area. La scelta di utilizzare tale livello è collegata al fatto che le attività di realizzazione ed esecuzione delle procedure di estrazione, trasformazione e caricamento avvengono in più stadi. In particolare, i risultati delle elaborazioni intermedie dei dati, solitamente le più complesse e costose in termini di risorse utilizzate, vengono memorizzati su un vero e proprio data base, ovvero la staging area STAIMPORT (Figura 6.20).



Figura 6.20: Utilizzo della Staging Area.

Nei casi in cui l'estrazione e la trasformazione dei dati non presenti particolari problemi, è possibile fare a meno di memorizzare i dati estratti in un'area temporanea, tuttavia, in generale, l'utilizzo della staging area è incentivato dal fatto che essa fornisce dei vantaggi significativi, tra cui indipendenza e integrazione tra sistemi e controlli di integrità e qualità dei dati.

Nel caso specifico, data la consistente quantità di dati e tabelle coinvolte nel processo di data warehousing e le numerose trasformazioni da apportare ai dati, è stato giudicato preferibile servirsene, utilizzandola come struttura intermedia tra le sorgenti ed il *data warehouse*.

6.6 Fase di estrazione

Questa fase è la prima del processo di ETL. In seguito all' identificazione delle tabelle del database operativo contenenti i dati necessari alle analisi, elencate nel quarto capitolo, si procede alla loro estrazione nella staging area (STAIMPORT), rendendole così disponibili nei flussi SSIS per la trasformazione e il caricamento. Lo schema seguito è pertanto il seguente:

- Estrazione dai server tramite query selettive ristrette alle tabelle e agli attributi interessanti;
- Memorizzazione nella staging area.

La Figura 6.21 mostra il flusso di controllo relativo all'estrazione dati dal server MSP, composto da una serie di Sequence Container, ciascuno dei quali contenente un Execute SQL Task e un Data Flow Task.

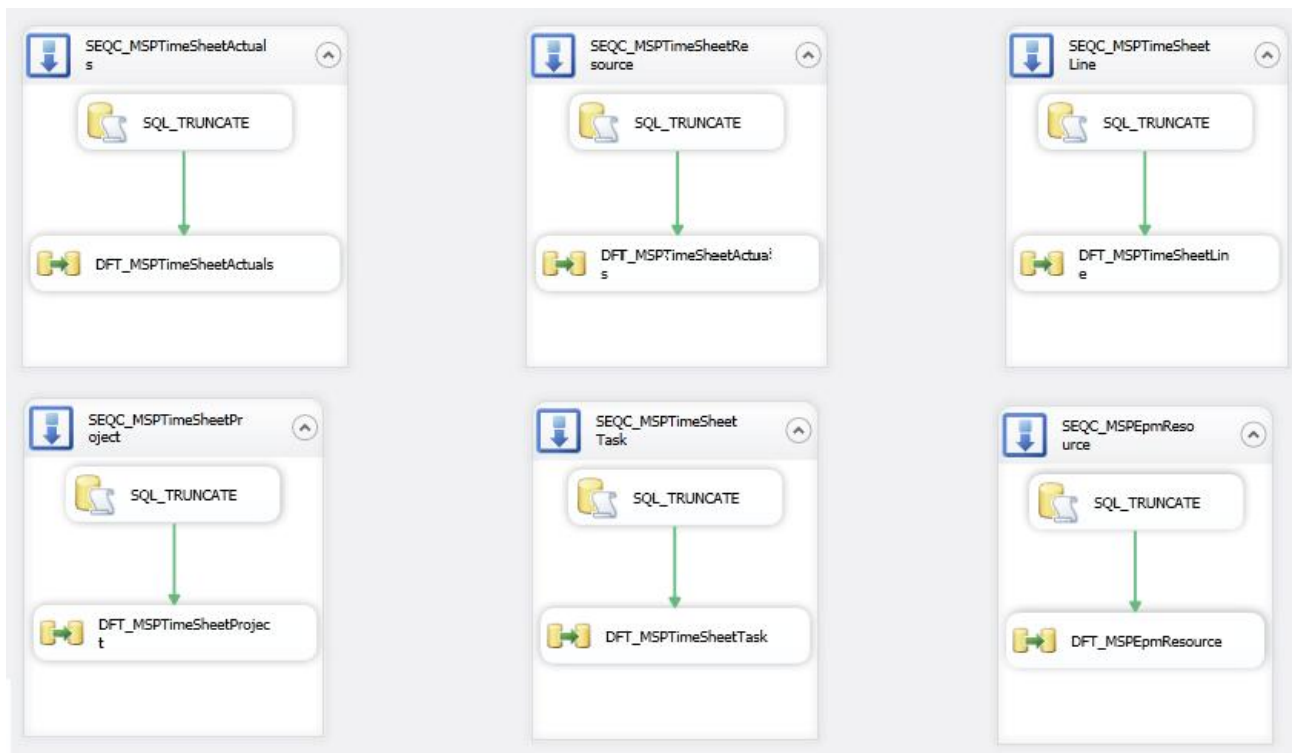


Figura 6.21: Flusso di controllo relativo all'estrazione dati dal server MSP

Per quanto riguarda l'estrazione dati dal server HPSC, il processo è analogo, con la differenza che, in attesa di risolvere il problema di interfacciamento con il server, le origini dati sono dei flat file statici. La creazione delle tabelle della staging area avviene automaticamente, grazie alle proprietà del componente destinazione di SSIS, il quale richiede che venga indicata una tabella di destinazione per il flusso dati, e, nel caso essa non esista fisicamente, la crea.

6.7 Trasformazione e Caricamento

In questa fase vengono opportunamente selezionati i dati utili, ripuliti, resi omogenei e integrati per creare le tabelle dimensionali e le tabelle dei fatti. In generale, il procedimento seguito prevede l'estrazioni dei dati dalla staging area, un insieme di operazioni effettuate nel flusso di dati e il caricamento nel *data warehouse*. Nel seguito, vengono dettagliati alcuni aspetti importanti riguardanti la creazione delle dimensioni e dei fatti.

- **Task.** L'attributo TaskGroupName, non presente nella base di dati operativa, viene ricavato a partire dall'attributo TaskName. Sotto esplicita richiesta del committente, è stata implementata una logica di assegnamento di ogni Task a un gruppo. La via più semplice per risolvere il problema sarebbe stata delegare ai dipendenti Accenture l'assegnazione manuale dei task al proprio gruppo di appartenenza tramite MDS, tuttavia questa soluzione è risultata impraticabile per via dell'elevatissimo numero di task presenti nella base di dati operativa. Un primo tentativo è stato effettuato tramite l'utilizzo del componente Fuzzy Grouping di SSIS, ma i risultati ottenuti sono stati abbastanza scadenti in quanto poco più del 30% dei task veniva assegnato alla categoria desiderata. Utilizzando il componente Script Component è stato pertanto implementato un algoritmo ad hoc di assegnamento in C#. La logica dell'algoritmo creato si basa su un confronto tra stringhe, utilizza le librerie del C# per ottenere l'elenco di gruppi dalla tabella di MDS dove sono memorizzati, fissa una soglia di tolleranza per l'affinità tra stringhe e assegna il task al gruppo che ottiene il punteggio maggiore. I test effettuati sull'efficienza dell'algoritmo sviluppato dimostrano che circa il 95% dei task ricade nel gruppo desiderato. Nella Figura 6.22 è mostrato il flusso dati con una finestra sul componente.

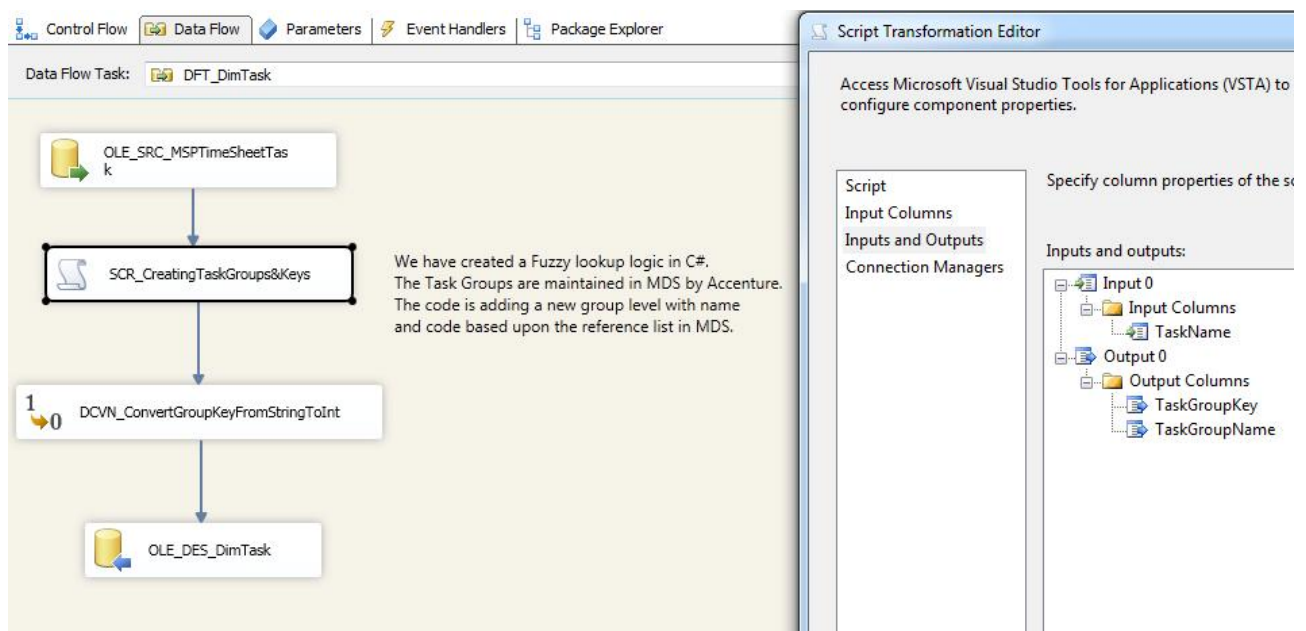


Figura 6.22: Flusso di dati della dimensione Task con finestra sullo Script Component.

Inoltre, l'ultimo attributo della dimensione task, ovvero MENumber, viene ricavato estraendo le prime 6 lettere dal nome del task (le quali, nel caso in cui il task faccia riferimento a un ticket di tipo Request, corrispondono esattamente al codice del task):

```
SELECT      TaskNameUID, TaskName, CASE WHEN Left(TaskName,6) like 'Q%' then
rtrim(Left(TaskName,6)) ELSE NULL END AS  ME_Number
FROM        MSP.MSP_TimeSheetTask
```

- **WBSType.** Come detto nel quarto capitolo, i dati necessari per la creazione di questa dimensione vengono prelevati da MDS.

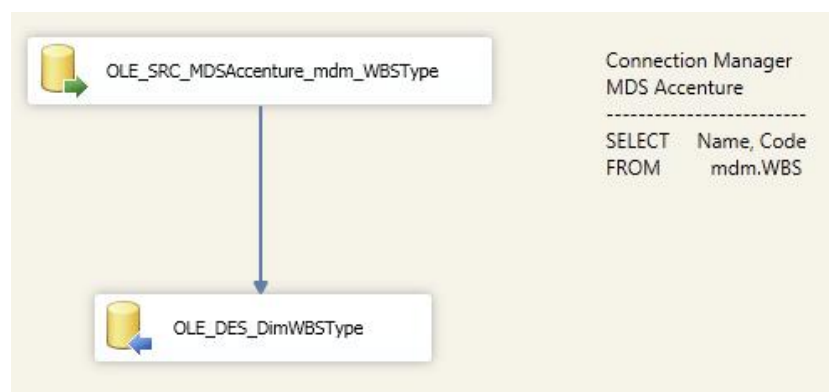


Figura 6.23: Flusso di dati della dimensione WBSType.

- **TicketStatus.** Le procedure di ETL per la creazione di questa dimensione sono assolutamente analoghe a quelle appena descritte per la dimensione precedente.

- **Projects.** In questo caso i dati necessari per la creazione della dimensione vengono estratti dalla staging area e integrati con altri dati prelevati da MDS.

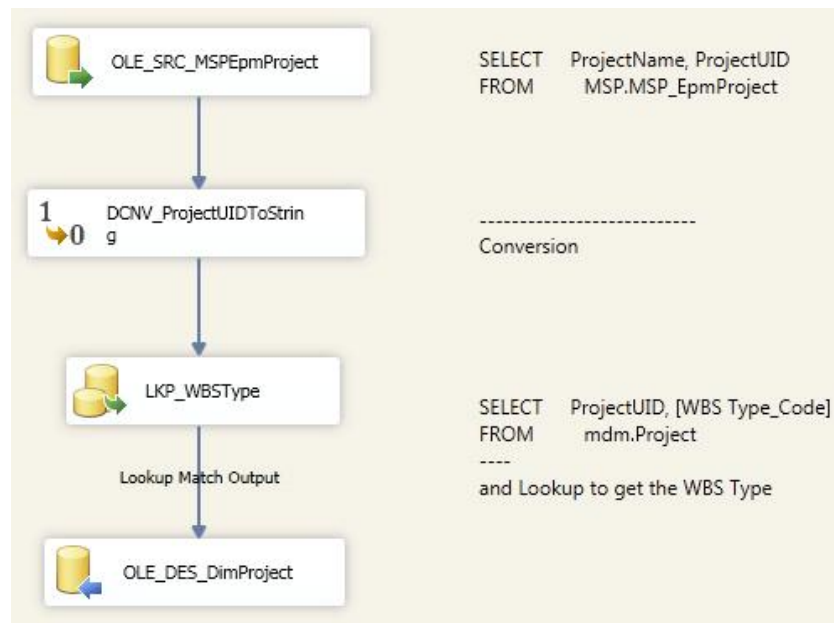


Figura 6.24: Flusso di dati della dimensione Projects.

- **TicketDetails.** Per la creazione di questa dimensione vengono da prima estratti i dati dalle tabelle precedentemente create nella staging area e successivamente vengono tra loro integrati e convertiti.

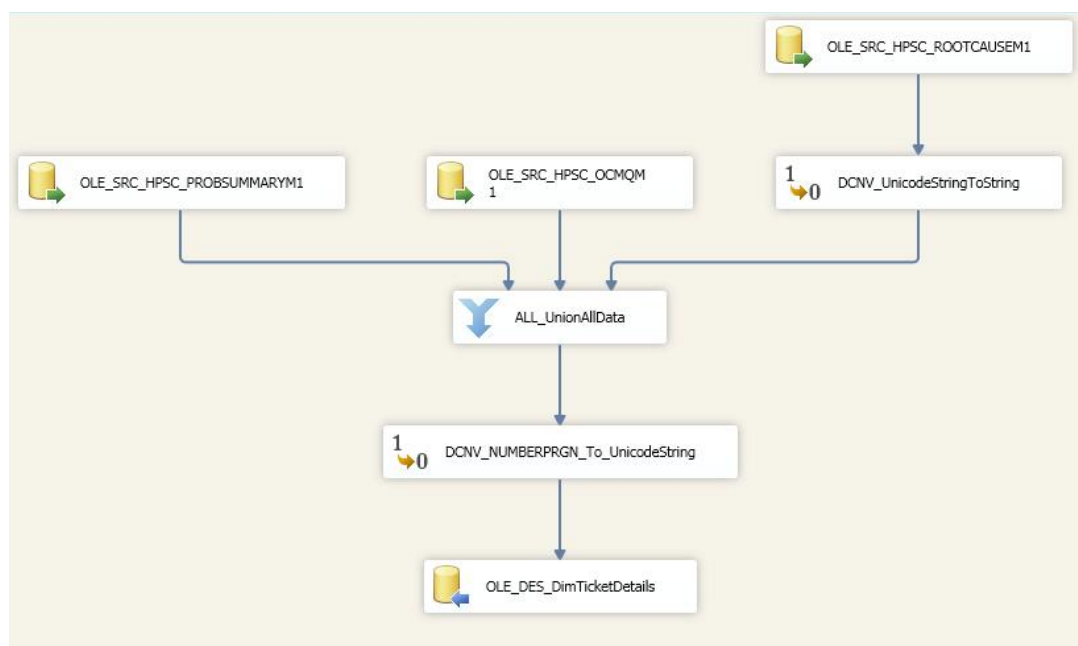


Figura 6.25: Flusso di dati della dimensione TicketDetails.

Infine l'attributo TicketType viene ricavato tramite uno script SQL inserito nel componente Execute SQL Task del flusso di controllo.

```
UPDATE [DW].[DW].[DimTicketDetails]
SET TicketType = CASE left([NUMBERPRGN],1)
                  WHEN 'I' then 'Incident'
                  when 'Q' then 'Request'
                  when 'P' then 'Problem'
ELSE NULL end
```

6.8 Slowly Changing Dimensions

In questa sezione viene presentata la strategia adottata per la gestione delle dimensioni a cambiamento lento. È stato stabilito di comune accordo col il Project Manager di non utilizzare il componente standard del flusso di dati Slowly Changing Dimension in quanto la versione per SQL Server 2012 è poco flessibile alle modifiche e poco intuitiva, nel senso che non permette di visualizzare a primo impatto tutte le logiche utilizzate per gestire il cambiamento e questo aspetto non si concilia con la politica aziendale sulla manutenzione del sistema descritta in precedenza. Per questa ragione è stato stabilito di creare la logica utilizzando una combinazione di altri componenti del flusso di dati. Nonostante ciò richieda un utilizzo di numerosi componenti, permette di poter modificare il flusso dati in diversi punti e operare ad hoc le eventuali modifiche da apportare.

Come specificato nel terzo capitolo, le dimensioni coinvolte nella gestione del cambiamento per le quali si applica il cambiamento Type 2 sono Organization e Resource. Nella dimensione Organization interessa applicare la logica del cambiamento Type 2 solo nel caso in cui venga modificato l'assegnamento di un sub team a un membro della gerarchia più elevato; nella dimensione Resource si applica invece nel caso in cui venga modificato uno dei seguenti attributi: Team_Code, Location_Code, Team_Name, Location_Name. Viene ora mostrata nel dettaglio la logica utilizzata per la dimensione Organization.

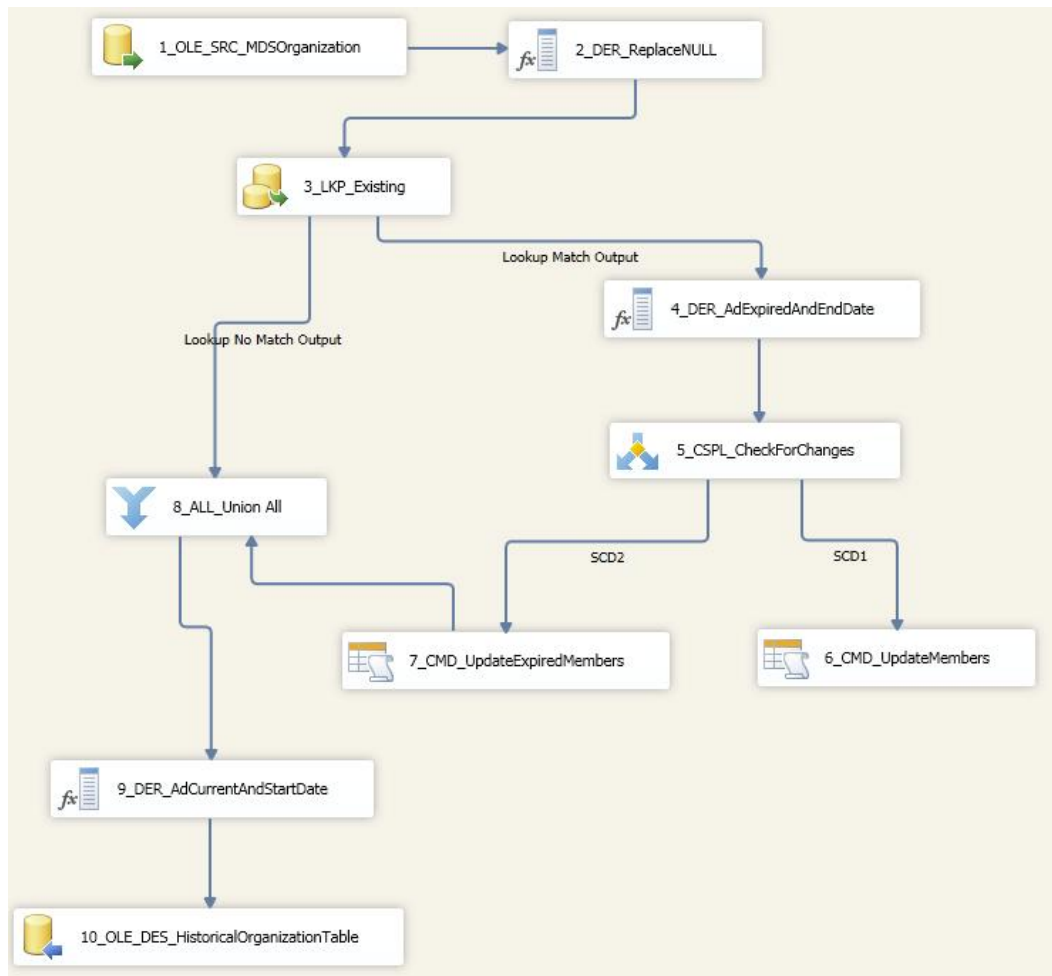


Figura 6.26: Flusso di dati che gestisce il cambiamento della dimensione Organization.

Partendo dai dati estratti da Master Data Services, si effettua una Lookup con la tabella storica estraendo solo le righe il cui attributo IsCurrent assume il valore numerico 1. Le righe che soddisfano il matching vengono a questo punto sottoposte a un controllo logico. La condizione sottoposta a verifica è la seguente:

```
(Organization_Code != Organization_Code_Existing) ||
(Tower_Code != Tower_Code_Existing) || (Team_Code_Existing != Team_Code)
```

Se verificata, si utilizza il componente OLE DB Command per tenere traccia del cambiamento:

```
UPDATE [SCD].[HistoricalOrganizationTable] SET [RowIsCurrent] = ?,
[RowValidTo] = ? WHERE [SubTeam_Code] = ? AND [RowIsCurrent] = 1
```


Se invece viene rilevata una modifica sul nominativo, viene lanciato un comando SQL sempre tramite il componente OLE DB Command per aggiornare le righe interessate:

```
UPDATE [SCD].[HistoricalOrganizationTable] SET [SubTeam_Name] = ?, [Team_Name] =  
?, [Tower_Name] = ?, [Organization_Name] = ? WHERE [SubTeam_Code] = ?  
AND [RowIsCurrent] = 1
```

Infine le righe che hanno subito una modifica fluiscono, insieme alle righe che non avevano soddisfatto la condizione di matching, fino al componente Derived Column, utilizzato in questo caso per inizializzare i campi RowIsCurrent e RowIsValidFrom.

In conclusione, nella creazione della dimensione vengono dapprima gestiti gli eventuali cambiamenti e successivamente viene eseguito, includendo nel flusso di dati la tabella storica, il package utilizzato per creare e mantenere aggiornata la dimensione. Per non dilungare eccessivamente la trattazione di questo argomento si omette di specificare i dettagli implementativi della dimensione Resource, dal momento che la logica utilizzata per gestire il cambiamento risulta del tutto analoga a quella appena enunciata.

6.9 Creazione delle tabelle dei fatti

In Figura 6.27 è mostrato il flusso di dati relativo alla creazione della tabella dei fatti Registrazione delle ore.

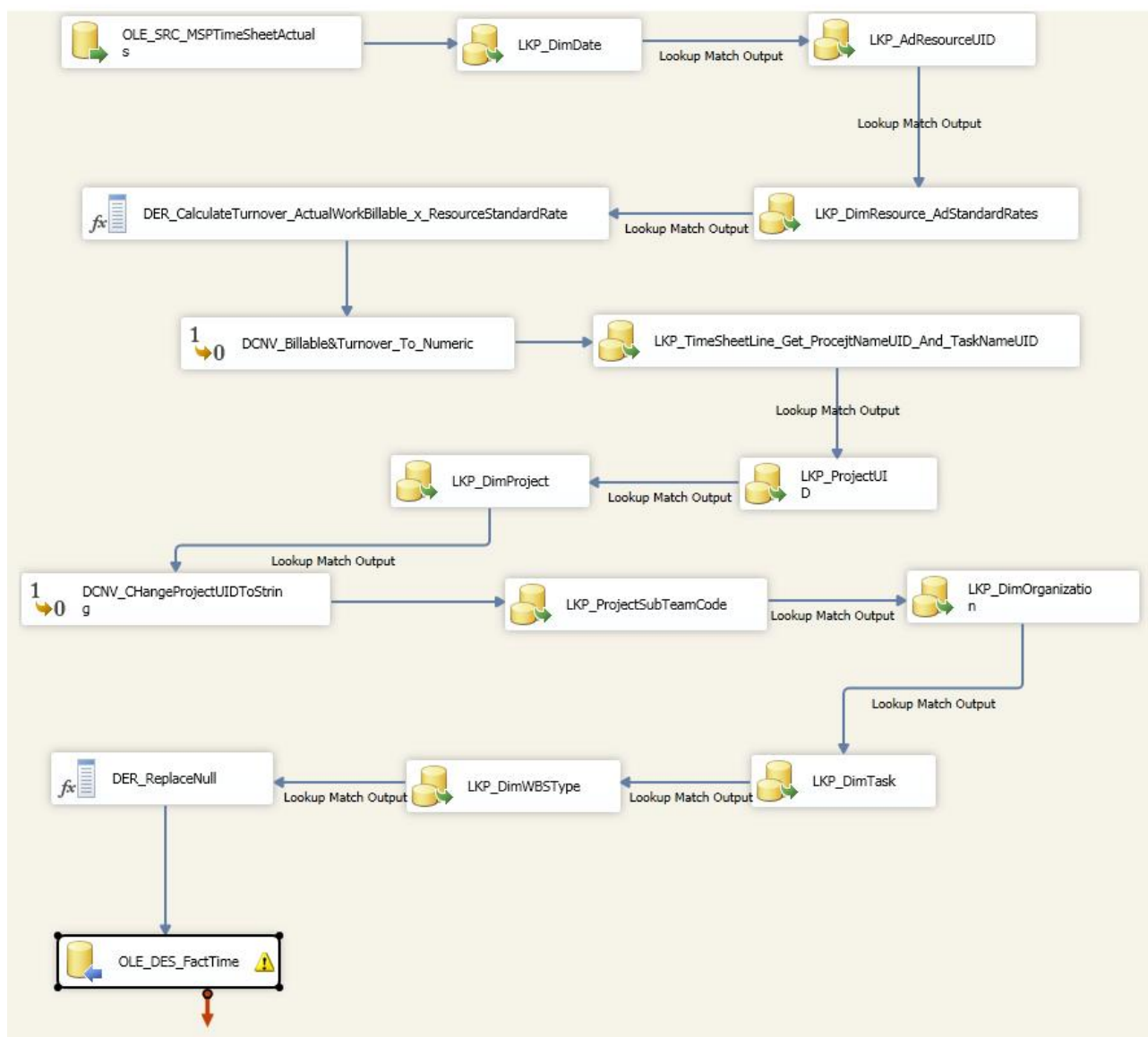


Figura 6.27: Flusso di dati relativo alla creazione della tabella dei fatti Registrazione delle ore.

Come si evince dalla Figura 6.27, il flusso integra la tabella operativa considerata Entità Evento con le dimensioni create e alcune tabelle operative importate nella staging area. Nel corso del flusso di dati vengono convertiti alcuni tipi di dati, effettuati dei calcoli e sostituiti gli eventuali valori NULL con il valore numerico -1 (in fase di creazione delle dimensioni viene inserito in

ciascuna di esse un elemento fittizio “Unknown”, la cui chiave è per l’appunto -1). Come accennato nella Sottosezione 5.2.3.1, le cattive performance riscontrate hanno portato a rivedere il flusso di dati relativo alla creazione delle tabelle dei fatti. La soluzione adottata consiste nell’effettuare i calcoli necessari alla creazione degli indicatori di performance in fase di ETL in modo da alleggerire le aggregazioni che verranno effettuate successivamente sul cubo. Per effettuare i calcoli è stato utilizzato il componente del flusso di dati Derived Column, come viene mostrato nella Figura 6.28.

BaseServiceOnSiteEffort	REPLACENULL(WBType_Code == "1" && Location_Code == "1" ? ActualWorkBillable / FTE_Hours : 0,0) < 0 ? 0 : REPLACENULL(WBType_Code == "1" && Location_Cod...
BaseServiceIDC_Effort	REPLACENULL(WBType_Code == "1" && Location_Code == "2" ? ActualWorkBillable / FTE_Hours : 0,0) < 0 ? 0 : REPLACENULL(WBType_Code == "1" && Location_Cod...
ME_OnSiteEffort	REPLACENULL(WBType_Code == "3" && Location_Code == "1" ? ActualWorkBillable / FTE_Hours : 0,0) < 0 ? 0 : REPLACENULL(WBType_Code == "3" && Location_Cod...
ME_IDC_Effort	REPLACENULL(WBType_Code == "3" && Location_Code == "2" ? ActualWorkBillable / FTE_Hours : 0,0) < 0 ? 0 : REPLACENULL(WBType_Code == "3" && Location_Cod...
CostME	REPLACENULL(WBType_Code == "3" ? [Turnover Actual] : 0,0)
ActualWorkME	REPLACENULL(WBType_Code == "3" ? ActualWorkBillable : 0,0)
UnassignedTime	REPLACENULL(TaskGroupKey == 57 ? ActualWorkBillable / FTE_Hours : 0,0) < 0 ? 0 : REPLACENULL(TaskGroupKey == 57 ? ActualWorkBillable / FTE_Hours : 0,0)
FTE	REPLACENULL(ActualWorkBillable / FTE_Hours,0) < 0 ? 0 : REPLACENULL(ActualWorkBillable / FTE_Hours,0)
InternalTime	REPLACENULL(WBType_Code == "2" ? ActualWorkBillable / FTE_Hours : 0,0) < 0 ? 0 : REPLACENULL(WBType_Code == "2" ? ActualWorkBillable / FTE_Hours : 0,0)
ProjectEffortOn-Shore	REPLACENULL(WBType_Code == "5" && Location_Code == "1" ? ActualWorkBillable / FTE_Hours : 0,0) < 0 ? 0 : REPLACENULL(WBType_Code == "5" && Location_Cod...
ProjectEffortOff-Shore	REPLACENULL(WBType_Code == "5" && Location_Code == "2" ? ActualWorkBillable / FTE_Hours : 0,0) < 0 ? 0 : REPLACENULL(WBType_Code == "5" && Location_Cod...
ActualWorkSpendOnBase...	REPLACENULL(WBType_Code == "1" && TaskGroupKey == 21 ? ActualWorkBillable : 0,0)
TimeSpendOnMEs	REPLACENULL(TicketType == "Request" && (REQUEST_TYPE == "Minor Enhancement" REQUEST_TYPE == "Fast Procurement") ? ActualWorkBillable : 0,0)

Figura 6.28: Dettaglio del componente Derived Column del flusso di dati utilizzato per effettuare dei calcoli preliminari sulla tabella dei fatti Registrazione delle ore.

Un procedimento del tutto analogo è stato adottato per la creazione della tabella dei fatti Ticket di Servizio.

TicketAge	DATEDIFF("dd",OpenTime,ReportingDate) < 0 ? NULL(DT_I1) : DATEDIFF("dd",OpenTime,ReportingDate)
OverdueAge	DATEDIFF("dd",Planned_End,ReportingDate) < 0 ? NULL(DT_I1) : DATEDIFF("dd",Planned_End,ReportingDate)
ResolvedThisPeriod	REPLACENULL(ReportingWeek == ResolvedWeek ? 1 : 0,0)
ResolvedInDueTime	REPLACENULL(DATEDIFF("dd",ResolvedTime,Planned_End) > 0 ? 1 : 0,0)
OverDue	REPLACENULL(DATEDIFF("dd",Planned_End,ReportingDate) > 0 ? 1 : 0,0)
VIP_Request	REPLACENULL(Critical_User == "true" ? 1 : 0,0)
DaysSinceLastUpdate	DATEDIFF("dd",LastUpdatedTime,ReportingDate) < 0 ? NULL(DT_I1) : DATEDIFF("dd",LastUpdatedTime,ReportingDate)
CheckOnDaysSinceLastU...	REPLACENULL((TicketType == "Request" && DATEDIFF("dd",LastUpdatedTime,ReportingDate) >= 30) (TicketType == "Inc...

Figura 6.29: Dettaglio del componente Derived Column del flusso di dati utilizzato per effettuare dei calcoli preliminari sulla tabella dei fatti Ticket di Servizio.

Sebbene ciò implichi un incremento dei tempi di esecuzione delle procedure di ETL e un maggiore utilizzo di spazio, i test effettuati sul cubo hanno evidenziato un netto miglioramento nelle performance di esecuzione. In particolare, nel caso delle aggregazioni più complesse, sono state

riscontrate differenze di performance che variano dalla decina di secondi ai minuti. Inoltre, sia l'incremento dei tempi di esecuzione delle procedure di ETL che la quantità di spazio aggiuntivo utilizzata risultano essere pressoché trascurabili.

6.10 Esecuzione procedure e frequenza di aggiornamento

Come accennato nelle Sottosezioni 2.5.1 e 2.5.2, il *data warehouse* verrà aggiornato a determinate cadenze temporali. A tale scopo è stato realizzato il package ExecutePackages, il quale, schedulato affinché venga eseguito a cadenza bisettimanale in corrispondenza con gli aggiornamenti del server MSP, esegue in background tutti i package necessari per l'aggiornamento di dimensioni e fatti. A causa delle problematiche di interfacciamento con il server HPSC riportate nella Sezione 4.5, è stato stabilito di attendere il momento in cui poter disporre di un interfacciamento diretto con il server prima di prendere una decisione definitiva in merito alla frequenza di aggiornamento del *data warehouse*. Tuttavia, in fase di riunione preliminare, anche in base a una stima sul volume giornaliero di dati immagazzinati nel server, era stato ritenuto ragionevole aggiornare quotidianamente il *data warehouse*. In ogni caso, il cubo potrà poi essere aggiornato in pochi click tramite Sql Server Management Studio. La Figura 6.30 mostra il flusso di controllo del package ExecutePackages.

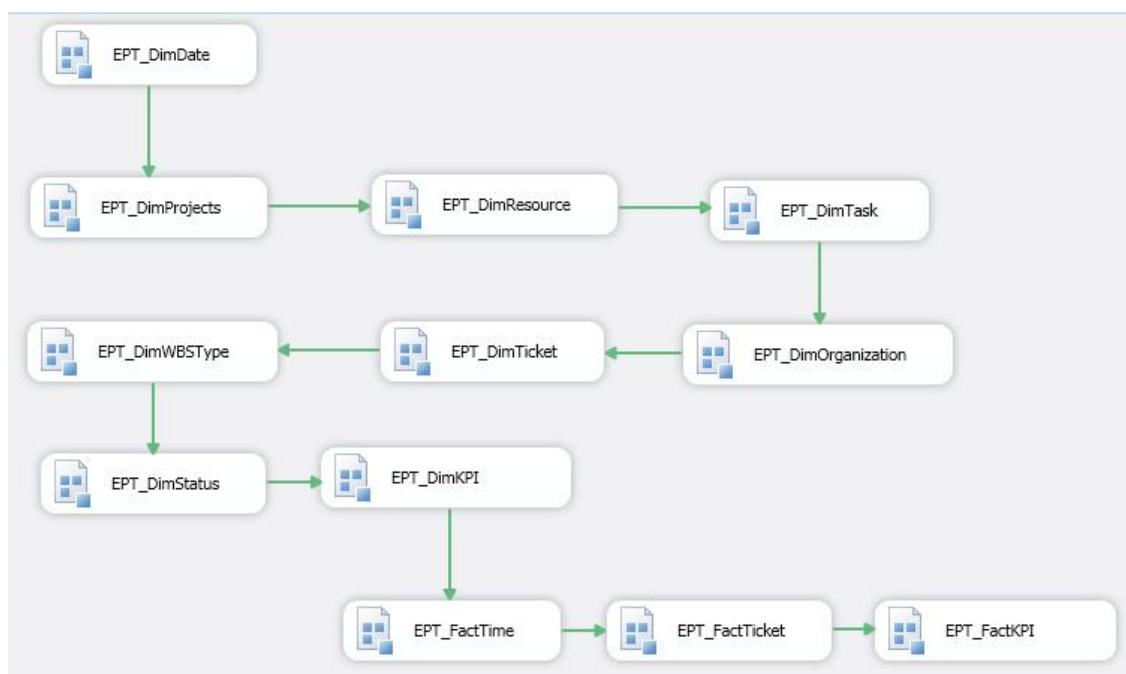


Figura 6.30: Flusso di controllo del pacchetto ExecutePackages.

Per concludere il capitolo si presentano dei diagrammi di flusso realizzati per riassumere il flusso di dati relativo ad alcune dimensioni la cui creazione comporta l'integrazione di dati provenienti da più origini.

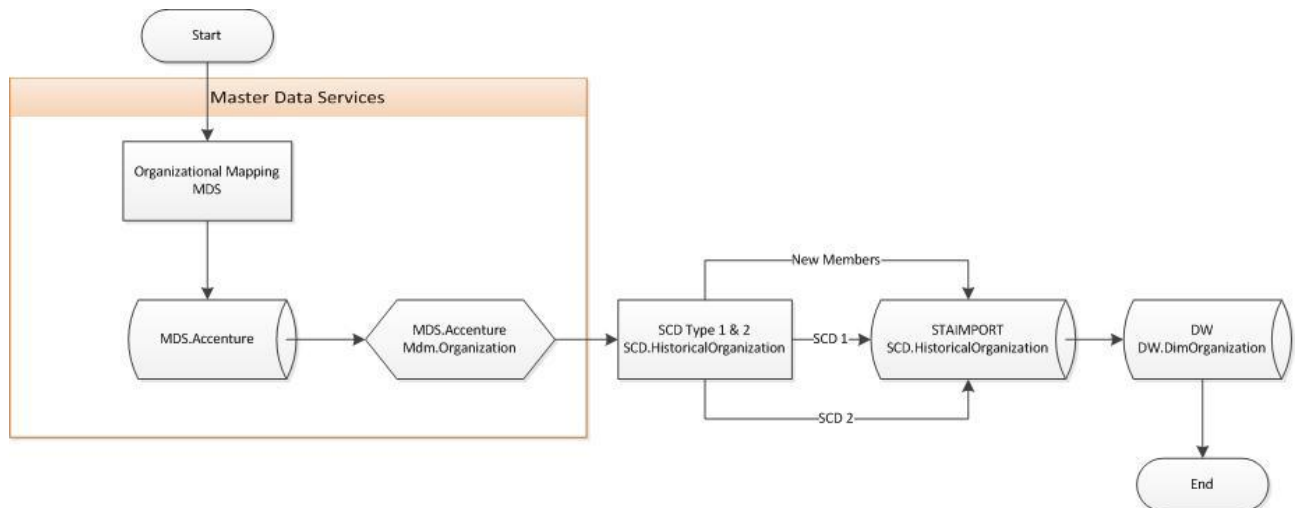


Figura 6.31: Flusso di dati relativo alla creazione della dimensione Organization.

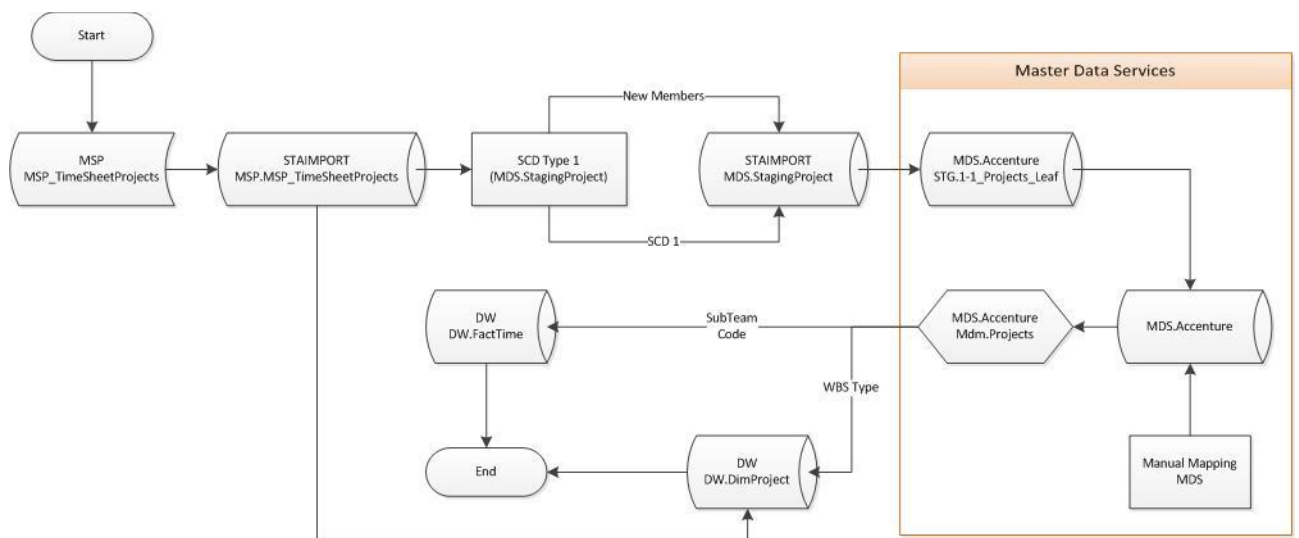


Figura 6.32: Flusso di dati relativo alla creazione della dimensione Project.

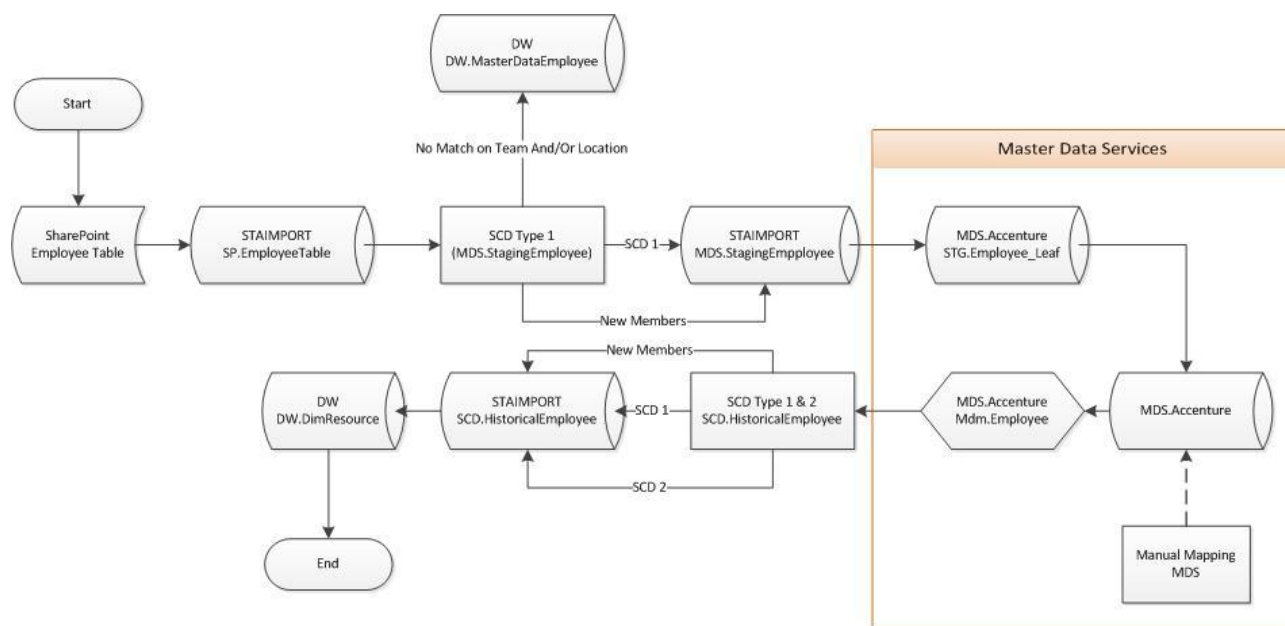


Figura 6.33: Flusso di dati relativo alla creazione della dimensione Resource.

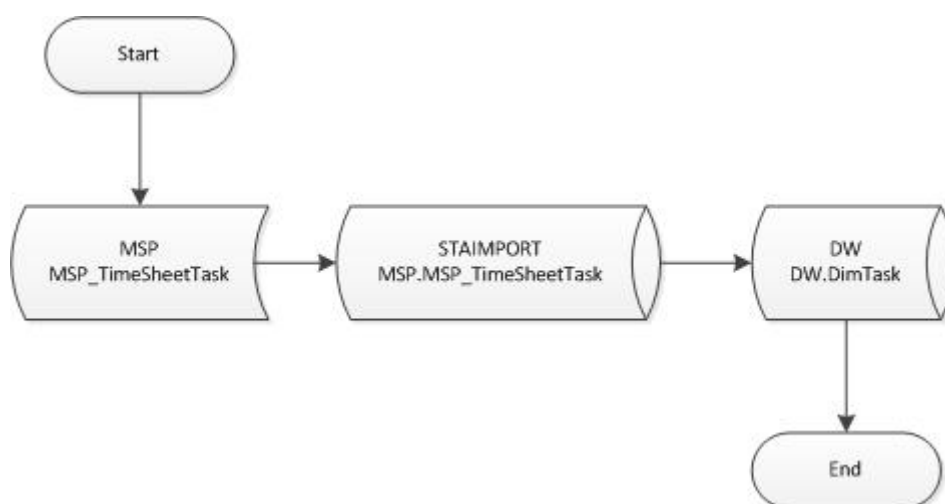


Figura 6.34: Flusso di dati relativo alla creazione della dimensione Task.

ESEMPI DI REPORTISTICA

Si descrive la parte finale del processo di data warehousing, ovvero la creazione dei report. Si presenta il software utilizzato per la realizzazione dei report, descrivendo le sue principali caratteristiche. Per concludere il capitolo, vengono messi a confronto i report utilizzati in precedenza con i report generati.

7.1 Reporting

Per agevolare l'interpretazione dei risultati delle analisi dei dati è fondamentale presentarli in modo opportuno. Come ricorda [Albano 12], esistono diversi strumenti per farlo, noti con il termine reporting tools. I modi più comuni sono:

- Rapporto tradizionale, che organizza il risultato con colonne di dati, intestazioni e uno o più livelli di dati di riepilogo parziali;
- Tabella a doppia entrata, come le tabelle pivot di Excel, che mostrano le aggregazioni delle misure rispetto ai valori delle dimensioni lungo gli assi cartesiani. Aggiungendo e togliendo dimensioni si ottengono gli effetti delle operazioni di drill down e roll-up;
- Grafici di natura diversa, come istogrammi, diagrammi lineari, diagrammi a torta.

La rappresentazione grafica del risultato spesso è essenziale per rendere il risultato comprensibile e utile a coloro che hanno bisogno di informazione di supporto alle decisioni, motivazione principale

dell'analisi multidimensionale. Tutti gli strumenti di reportistica consentono sia di formulare un'analisi dei dati senza scrivere l'interrogazione in SQL, sia di produrre una rappresentazione grafica del risultato. Alcuni DBMS consentono di produrre una rappresentazione grafica del risultato anche con analisi espresse in SQL.

Come ampiamente discusso nei capitoli introduttivi, lo scopo principale del progetto è la completa automazione della reportistica legata al sistema informativo relativo alla collaborazione tra Accenture e Velux. Per quanto riguarda l'automazione, è stata realizzata tramite la creazione del *data warehouse* dettagliatamente descritto nei capitoli precedenti. Per generare la reportistica, invece, sono stati utilizzati gli strumenti Microsoft Excel e SQL Server Reporting Services.

7.2 SQL Server Reporting Services (SSRS)

SQL Reporting Services è una piattaforma di report basata su server che fornisce una vasta gamma di strumenti e servizi pronti all'uso che consentono di creare, distribuire e gestire report per l'organizzazione, nonché programmare funzionalità che permettono di estendere e personalizzare la funzione di creazione di report. SSRS offre una soluzione di reporting completa che include tradizionali report gestiti, report interattivi per gli utenti finali e report incorporati. Gli utenti possono effettuare sottoscrizioni ai report desiderati o possono facilmente accedere su richiesta ai report con le informazioni più recenti. Il tool include Report Builder, uno strumento di reporting ad hoc che aiuta a connettere in modo protetto gli utenti aziendali ai dati, in modo da consentire loro di esplorare i dati relazionali e di creare e aggiornare i report con facilità.

Come riportato da [Msdn 13], SSRS è fornito di un set completo di strumenti per creare, gestire e recapitare report, nonché API che consentono agli sviluppatori di integrare o estendere l'elaborazione di dati e report in applicazioni personalizzate. Gli strumenti a disposizione funzionano all'interno dell'ambiente Microsoft Visual Studio e sono completamente integrati con i componenti di SQL Server. Per quanto concerne le tipologie di report che è possibile creare, la piattaforma consente di creare report interattivi, tabulari, grafici o in formato libero da origini dati relazionali, multidimensionali o basati su XML, i quali possono includere una visualizzazione dei dati dettagliata, con grafici, mappe e grafici sparkline e possono essere pubblicati, pianificati o visualizzati su richiesta.

[Msdn 13] evidenzia come Reporting Services consenta di scegliere tra un'ampia gamma di formati di visualizzazione, di esportare i report in altre applicazioni, ad esempio Microsoft Excel, e

di sottoscrivere report pubblicati. I report creati possono essere visualizzati tramite una connessione basata su Web o come parte di un'applicazione Microsoft Windows o di un sito di SharePoint. È inoltre possibile creare avvisi dati nei report pubblicati in un sito di SharePoint e notificare i soggetti interessati per mezzo di messaggi di posta elettronica quando i dati del report cambiano.

7.3 Vantaggi dell'utilizzo di SQL Server Reporting Services (SSRS)

Oltre alla già citata partnership tra D60 e Microsoft, SSRS è stato scelto per la sua semplicità di utilizzo e per una serie di vantaggi che offre in termini di visualizzazione dati, origine dati e progettazione. Vengono elencati quelli che secondo [Msdn 13] risultano essere i più significativi:

- **Visualizzazione dati.** SSRS permette di visualizzare i dati in diverse modalità interconnesse utilizzando le aree dati, di organizzarli in tabelle, matrici o schede-oblique, di espandere e comprimere gruppi, grafici, misuratori, indicatori e mappe, con la possibilità di annidare i grafici nelle tabelle. SSRS permette inoltre di visualizzare report per utilizzo personale oppure pubblicarli su un server di report o un sito di SharePoint per condividerli con il team o l'organizzazione. È possibile esportare il report in più formati di file oppure inviarlo ai sottoscrittori come posta elettronica o sottoforma di un file condiviso e creare vari report collegati in cui si applicano diversi set di parametri.
- **Origine dati.** SSRS consente di utilizzare origini dati condivise, query condivise e sottoreport per definire modalità di visualizzazione dei dati utilizzabili più volte. È possibile inoltre gestire le origini dati del report separatamente dalla definizione di quest'ultimo. Ad esempio, si può convertire un'origine dati di prova in un'origine dati di produzione senza modificare il report.
- **Progettazione.** SSRS permette di progettare report in un layout con formato libero, ovvero non limitato per settori di informazioni. È possibile organizzare la visualizzazione dei dati sulla pagina in modo da favorire comprensione, analisi e azione, abilitare azioni di drill-through e di ordinamento, descrizioni comando e parametri di report per consentire al lettore di interagire con il report. Per quanto concerne l'utilizzo dei parametri e delle espressioni, si possono utilizzare parametri combinati con espressioni personalizzate per consentire ai lettori di controllare il modo in cui i dati vengono filtrati, raggruppati e ordinati.

7.4 I report iniziali

Come descritto nei capitoli introduttivi di questo documento, gran parte della reportistica, il cui fulcro è costituito dai valori espressi dagli indicatori di performance, è realizzata manualmente da 9 dipendenti di Accenture operanti in diverse postazioni sparse tra l'India e le Filippine e coordinati dal responsabile di progetto operante nella sede danese. La gestione della reportistica precedente era affidata principalmente al reparto ADU, il quale generava più di 200 report servendosi dello strumento "Crystal Reports", tra cui anche report manuali, orientati soprattutto al monitoraggio e all'analisi interna. Uno dei principali report manuali è rappresentato dall'analisi dei KPI. In Figura 7.1 viene mostrato, a titolo di esempio, la struttura di quest'ultimo report manuale, creato utilizzando Microsoft Excel e relativo al livello gerarchico Tower "AO".

Home									
AO									
Sustainable Profitability									
Metric Name	Unit	Desired Trend	Target	RAG Status	Current Period	Prior Period	Variance to target	4 Weeks Average	Trend 4 weeks
7.1 Base Service On-Site Effort	FTE	↓	17,2	✓	13,5	10,5	21%	14,1	↑
8.1 Base Service IDC Effort	FTE	↓	61,1	✓	49,4	50,1	19%	46,0	↓
9.1 ME On-Site Effort	FTE	↑	12,1	✗	10,7	11,9	-11%	11,1	↔
10.1 ME IDC Effort	FTE	↑	8,0	✓	8,7	7,4	9%	6,5	↑
11.1 IDC Contribution (ME-Effort)	%	↑	40,0%	✓	44,8%	38,2%	12%	36,5%	↔
13.1 Average Hourly Rate per ME	DKK	↓	565,0	✓	520,0	568,5	8%	577,3	↑
14.1 Unassigned Time	FTE	↓	2,5	✗	4,3	3,9	-71%	2,9	↓
15.1 Internal Time	%	↓	0,1	✓	0,0	0,0	55%	0,0	↔
16.1 Project Effort On-Shore	FTE	↑	0,0	N/A	15,2	15,6	N/A	15,7	↑
44.1 Project Effort Off-Shore	FTE	↑	0,0	N/A	11,2	8,6	N/A	8,3	↑
Operational Excellence									
Metric Name	Unit	Desired Trend	Target	RAG Status	Current Period	Prior Period	Variance to target	4 Weeks Average	Trend 4 weeks
17.1 Incidents Resolved in Due Time	%	↑	82%	✓	98,8%	97,0%	21%	98,2%	↔
18.1 Overdue Incidents	#	↓	75	✓	27,0	24,0	64%	26,5	↑
19.1 Average Ageing (Incidents)	Days	↓	60	✗	128,1	141,1	-114%	128,1	↓
20.1 Average Time Spend (Incidents)	Hours	↓	20	✓	4,3	5,8	79%	N/A	N/A
21.1 ME's Resolved in Due Time	%	↑	95%	✗	79,4%	89,7%	-16%	84,6%	↔
22.1 Overdue ME's	#	↓	80	✓	66	63	18%	54	↓
23.1 Average Ageing (ME)	Days	↓	45	✓	41	42	9%	58	↑
Metric Name	Unit	Desired Trend	Target	RAG Status	Current Period	Prior Period	Variance to target	4 Weeks Average	Trend 4 weeks
24.1 Average Time Spend (ME)	Hours	↓	12,0	✓	11,1	10,9	8%	12,1	↓
25.1 ME Backlog	Hours	↓	0,0	N/A	7493,9	6878,0	N/A	7040,4	↓
26.1 Broken LT1	%	↓	10%	✓	2%	7%	78%	7%	↔
27.1 Budget Overruns (%)	%	↓	10%	✗	24%	10%	-138%	19%	↔
28.1 Average Budget Overrun	Hours	↓	0,0	N/A	9,6	10,1	N/A	9,9	↔
42.1 >30 Days Since Last Update	#	↓	7,0	✗	111,0	115,0	-1486%	114,8	↓
43.1 >7 Days Since Last Update	#	↓	7,0	✗	41,0	37,0	-486%	55,5	↓
44.1 Open Problems	#	↓	0	N/A	5	12	N/A	40	↑
45.1 Overdue Problems	#	↓	0	N/A	0	0	N/A	5	↓

Figura 7.1: Accenture KPI, livello gerarchico Tower.

In Figura 7.2 viene invece mostrato un report relativo al livello Team (Organization “ADU”, Tower “IO”, Team “License Management”), anch’esso realizzato manualmente tramite Microsoft Excel.

	#	Description	Unit	Frequency	RAG Status	Target	Desired Trend	Current Period	Variance to Target	Prior Period	4 Weeks Average	4 Weeks Trend	Comments
Best People													
Sustainable Profitability	7.2	Base Service On-Site Effort	FTE	W	N/A	0,00	↓	0,00	N/A	0,00	0,00	→	
	8.2	Base Service IDC Effort	FTE	W	N/A	0,00	↓	0,00	N/A	0,00	0,00	→	
	9.2	ME On-Site Effort	FTE	W	N/A	0,00	↑	0,00	N/A	0,00	0,00	→	
	10.2	ME IDC Effort	FTE	W	N/A	0,00	↑	0,00	N/A	0,00	0,00	→	
	11.2	IDC Contribution (ME Effort)	%	W	N/A	0%	↑	No data	N/A	No data	N/A	N/A	
	12.2	IDC Contribution (ME Resolved)	%	W	🔴	50%	↑	0%	-100%	0%	0%	→	
	13.2	Average hourly Rate per ME	DKK	W	N/A	650,00	↓	No data	N/A	No data	N/A	N/A	
	14.2	Unassigned Time	FTE	W	N/A	0,00	↓	0,00	N/A	0,00	0,00	→	
	15.2	Internal Time	%	W	N/A	0,0%	↓	0,0%	N/A	0,0%	0,0%	→	
	16.2	Project Effort On-Shore	FTE	W	N/A	0,00	↑	0,00	N/A	0,00	0,00	→	
Operational Excellence	44.2	Project Effort Off-Shore	FTE	W	N/A	0,00	↑	0,00	N/A	0,00	0,00	→	
	17.2	Incidents Resolved in Due Time	%	W	N/A	82%	↑	No data	N/A	No data	N/A	N/A	
	18.2	Overdue Incidents	#	W	N/A	0,00	↓	0,00	N/A	0,00	0,00	→	
	19.2	Average Ageing (Incidents)	Days	W	N/A	0,00	↓	No data	N/A	No data	N/A	N/A	
	20.2	Average Time Spend (Incidents)	Hours	W	N/A	0,00	↓	No data	N/A	No data	N/A	N/A	
	21.2	ME's Resolved in Due Time	%	W	N/A	95%	↑	No data	N/A	No data	N/A	N/A	
	22.2	Overdue ME's	#	W	N/A	0,00	↓	0,00	N/A	0,00	0,00	→	
	23.2	Average Ageing (ME)	Days	W	N/A	0,00	↓	No data	N/A	No data	N/A	N/A	
	24.2	Average Time Spend (ME)	Hours	W	N/A	0,00	↓	No data	N/A	No data	N/A	N/A	
	25.2	ME Backlog	Hours	W	N/A	0,00	↓	0,00	N/A	0,00	0,00	→	
Operational Excellence	26.2	Broken LT1	%	W	N/A	10%	↓	No data	N/A	No data	N/A	N/A	
	27.2	Budget Overruns (%)	%	W	N/A	5%	↓	No data	N/A	No data	N/A	N/A	
	28.2	Average Budget Overrun	Hours	W	N/A	0,00	↓	0,00	N/A	0,00	0,00	→	
	42.2	>30 Days Since Last Update (ME)	#	W	N/A	0,00	↓	0,00	N/A	0,00	0,00	→	
	43.2	>7 Days Since Last Update (Incidents)	#	W	N/A	0,00	↓	1,00	N/A	0,00	0,75	→	
	44.2	Open Problems	#	W	N/A	0	↓	0	N/A	0	0	→	
	45.2	Overdue Problems	#	W	N/A	0	↓	0	N/A	0	0	→	

Figura 7.2: Accenture KPI, livello gerarchico Team.

Infine, in Figura 7.3 viene mostrato un report relativo al livello gerarchico Subteam. Il report include nel lato sinistro i dettagli del Subteam (Subteam “Teklogix” del Team “Service Management”) e riepiloga nel lato destro i dati del livello superiore (Team).

Home		Team Lead Scorecard Service Management																		
	#	Description	Unit	Frequency	Target	Current Period	Variance to Target	Prior Period	4 Weeks Average	4 Weeks Trend	Teklogix	RAG Status	Target	Desired Trend	Current Period	Variance to Target	Prior Period	4 Weeks Average	4 Weeks Trend	Comments
Sustainable Profitability	7.2	Base Service On-Site Effort	FTE	W	0,40	0,05	🟢	0,05	0,03	→	88%	N/A	0,00	↓	9,28	N/A	9,55	10,80	↑	
	8.2	Base Service IDC Effort	FTE	W	3,40	0,00	🟢	0,00	0,00	→	100%	N/A	0,00	↓	5,99	N/A	5,62	5,79	→	
	9.2	ME On-Site Effort	FTE	W	0,00	0,00	N/A	0,00	0,38	→	N/A	N/A	0,00	↑	0,21	N/A	0,10	0,08	→	
	10.2	ME IDC Effort	FTE	W	0,00	0,00	N/A	0,00	0,14	→	N/A	N/A	0,00	↑	0,00	N/A	0,00	0,00	→	
	11.2	IDC Contribution (ME Effort)	%	W	0%	No data	N/A	No data	38%	N/A	N/A	N/A	0%	↑	0%	N/A	0%	0%	→	
	12.2	IDC Contribution (ME Resolved)	%	W	50%	0%	🔴	0%	0%	→	-100%	🔴	50%	↑	0%	-100%	0%	0%	→	
	13.2	Average hourly Rate per ME	DKK	W	650,00	No data	N/A	No data	609,27	N/A	N/A	🔴	650,00	↓	758,13	-0,17	780,63	769,38	→	
	14.2	Unassigned Time	FTE	W	0,00	0,00	N/A	0,00	0,00	→	N/A	N/A	0,00	↓	0,00	N/A	0,00	0,00	→	
	15.2	Internal Time	%	W	0,0%	0,0%	N/A	0,0%	0,0%	→	N/A	N/A	0,0%	↓	0,0%	N/A	0,1%	0,7%	→	
	16.2	Project Effort On-Shore	FTE	W	0,00	0,00	N/A	0,00	0,00	→	N/A	N/A	0,00	↑	0,23	N/A	0,23	0,21	→	
44.2	Project Effort Off-Shore	FTE	W	0,00	0,00	N/A	0,00	0,00	→	N/A	N/A	0,00	↑	0,00	N/A	0,00	0,00	→		
Operational Excellence	17.2	Incidents Resolved in Due Time	%	W	82%	No data	N/A	100%	N/A	→	N/A	🟢	82%	↑	89%	8%	100%	N/A	N/A	
	18.2	Overdue Incidents	#	W	0,00	0,00	N/A	0,00	0,00	→	N/A	N/A	0,00	↓	4,00	N/A	4,00	4,25	→	
	19.2	Average Ageing (Incidents)	Days	W	0,00	No data	N/A	No data	N/A	N/A	→	N/A	0,00	↓	180,00	N/A	155,00	162,83	↓	
	20.2	Average Time Spend (Incidents)	Hours	W	0,00	No data	N/A	0,00	N/A	→	N/A	N/A	0,00	↓	12,00	N/A	5,74	8,43	↓	
	21.2	ME's Resolved in Due Time	%	W	95%	No data	N/A	No data	100%	N/A	→	🔴	95%	↑	0%	-100%	No data	0%	N/A	
	22.2	Overdue ME's	#	W	0,00	0,00	N/A	0,00	0,00	→	N/A	N/A	0,00	↓	4,00	N/A	6,00	4,50	↓	
	23.2	Average Ageing (ME)	Days	W	0,00	No data	N/A	No data	N/A	N/A	→	N/A	0,00	↓	65,75	N/A	149,17	158,35	↓	
	24.2	Average Time Spend (ME)	Hours	W	0,00	No data	N/A	No data	4,75	N/A	→	N/A	0,00	↓	28,00	N/A	No data	28,00	N/A	
	25.2	ME Backlog	Hours	W	0,00	4,50	N/A	1,50	353,75	↓	N/A	N/A	0,00	↓	106,90	N/A	117,40	81,95	↓	
	26.2	Broken LT1	%	W	10%	No data	N/A	No data	0%	N/A	→	N/A	10%	↓	No data	N/A	0%	0%	N/A	
	27.2	Budget Overruns (%)	%	W	5%	No data	N/A	No data	0%	N/A	→	🟢	5%	↓	0%	100%	No data	0%	N/A	
	28.2	Average Budget Overrun	Hours	W	0,00	4,00	N/A	4,00	1,00	→	N/A	N/A	0,00	↓	2,25	N/A	2,25	2,23	→	
	42.2	>30 Days Since Last Update (ME)	#	W	0,00	3,00	N/A	3,00	2,75	↓	N/A	N/A	0,00	↓	9,00	N/A	11,00	11,00	↓	
	43.2	>7 Days Since Last Update (Incidents)	#	W	0,00	2,00	N/A	1,00	0,00	→	N/A	N/A	0,00	↓	14,00	N/A	0,00	13,25	↓	
44.2	Open Problems	#	W	0	0	N/A	0	0	→	N/A	N/A	0	↓	1	N/A	0	2	→		
45.2	Overdue Problems	#	W	0	0	N/A	0	0	→	N/A	N/A	0	↓	0	N/A	0	0	→		

Figura 7.3: Accenture KPI, livello gerarchico SubTeam.

7.5 I report realizzati

Uno degli obiettivi principali del progetto era che tutti i report generati in precedenza con Crystal Reports e con Excel venissero modificati prelevando i dati da una nuova origine dati o ricreati nei nuovi strumenti di reportistica. In particolare, era stato preventivato che la nuova soluzione dovesse supportare report permanenti, report ad hoc/pivot per le analisi dinamiche e ovviamente permettesse di analizzare i valori dei KPI confrontandoli con quelli dei target. La soluzione avrebbe dovuto consentire di godere di una maggiore accessibilità alle analisi ad hoc per gli utenti finali e soprattutto della più completa automazione, offrendo al contempo qualità, tracciabilità e accuratezza.

Tutti gli obiettivi prefissati sono stati raggiunti utilizzando Microsoft Excel e SSRS. Tramite Microsoft Excel sono stati realizzati dei report ad hoc servendosi delle tabelle pivot connesse al cubo (o, opzionalmente, alla base di dati del modello tabulare), mentre tramite SSRS, utilizzando a seconda delle esigenze i vari strumenti grafici e testuali di cui esso dispone (Text-based query designer, Graphical query designer, Report model query designer e MDX query designer), sono stati ricreati i report per le analisi sui KPI. Vengono mostrati, a titolo di esempio, alcuni report realizzati tramite SSRS. Il primo mostra l'analisi settimanale dei KPI relativi al livello Organization "ADU" e al livello Tower "AO". Ciascun indicatore di performance corrisponde a un preciso requisito di analisi descritto nel secondo capitolo di questo documento, sebbene la numerazione visualizzata nei report faccia riferimento, per ovvie ragioni, alla numerazione interna stabilita da Accenture e non quella indicata nelle Sezioni 2.4.1 e 2.4.2. Come si evince dalla Figura 7.2, sono stati ricreati tutti i campi presenti nel report mostrato in Figura 7.1, con l'aggiunta della possibilità di visualizzare, per ogni KPI, un grafico cartesiano indicante la variazione del KPI rispetto al trend al variare delle settimane (Figura 7.3).

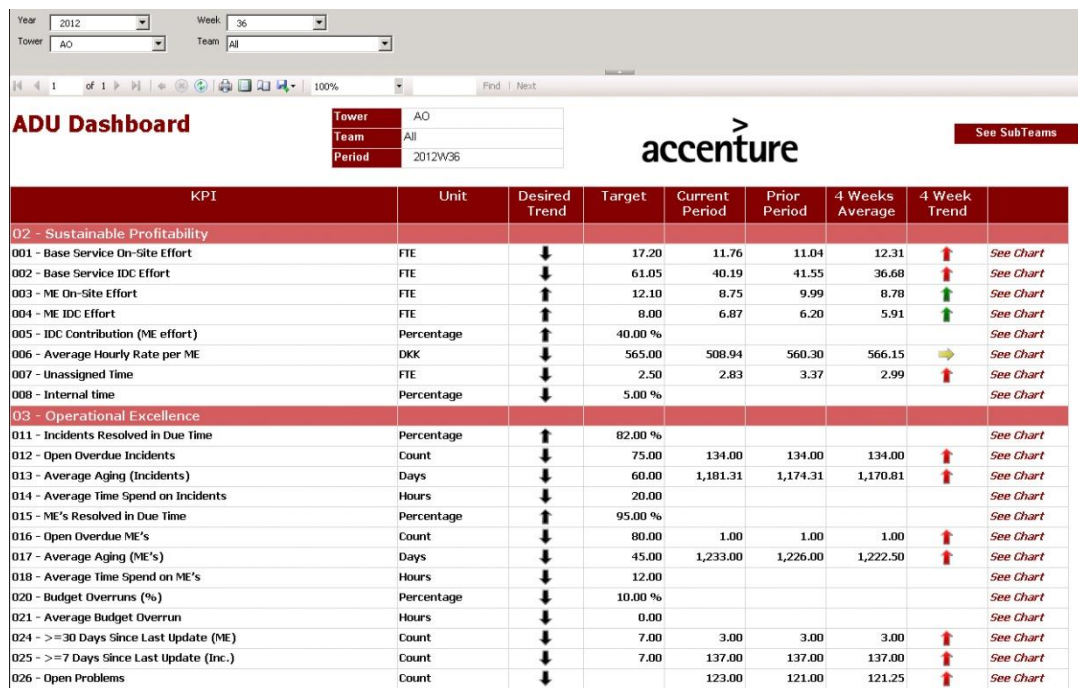


Figura 7.4: Accenture ADU Dashboard realizzata tramite SSRS.

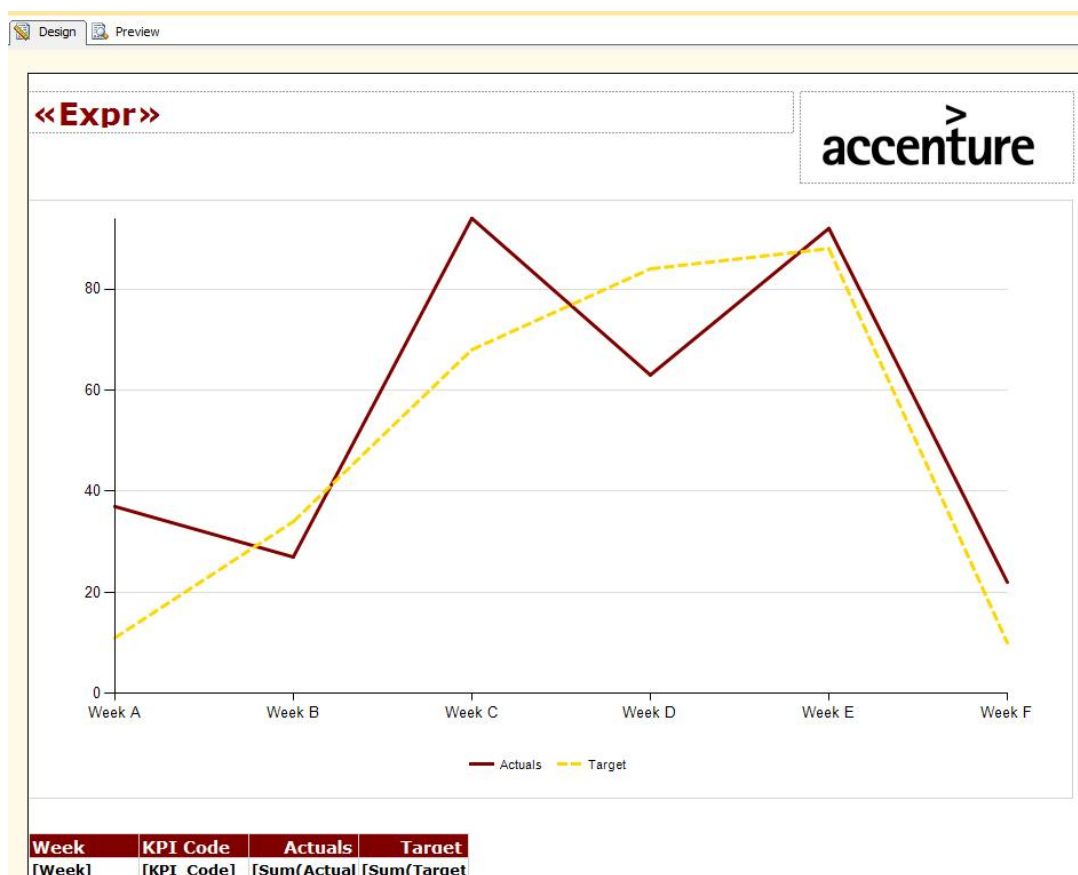


Figura 7.5: Grafico che mostra il trend settimanale di un generico KPI, realizzata tramite SSRS.

La Figura 7.6, relativa al design di un generico report a livello Tower, evidenzia invece un'altra modifica rispetto ai report iniziali, richiesta dal committente e realizzata tramite gli strumenti di SSRS, ovvero il menu laterale di navigazione. A differenza dei report iniziali che mostravano nella stessa schermata sia i dettagli del singolo Team che le aggregazioni su tutti i Team appartenenti allo stesso livello Tower, il collegamento "Go to Teams report" che appare nel lato destro della Figura 7.6 consente di navigare tra i livelli della gerarchia senza appesantire il report a livello grafico.

ADU Dashboard

Tower

Period

@OrganizationTowerName.Label

@DateWeek.Label

>

accenture

Go to Teams report

KPI	Unit	Desired Trend	Target	RAG Status	Current Period	Prior Period	4 Weeks Average	4 Weeks Trend	
[KPI_Group]									
[KPI]	[Unit]	↑	[Sum(Targe	🟢	[Sum(Actual	[Sum(Prior	[Sum(ID4 1	📉	Go to Chart

Figura 7.6: Struttura di un report di livello Tower realizzata tramite SSRS.

Viene infine mostrata la schermata di un report di livello Team ("Service Management"), che, seguendo la stessa logica di navigazione dei livelli superiori, consente agevolmente di visualizzare i dettagli del livello gerarchico successivo.

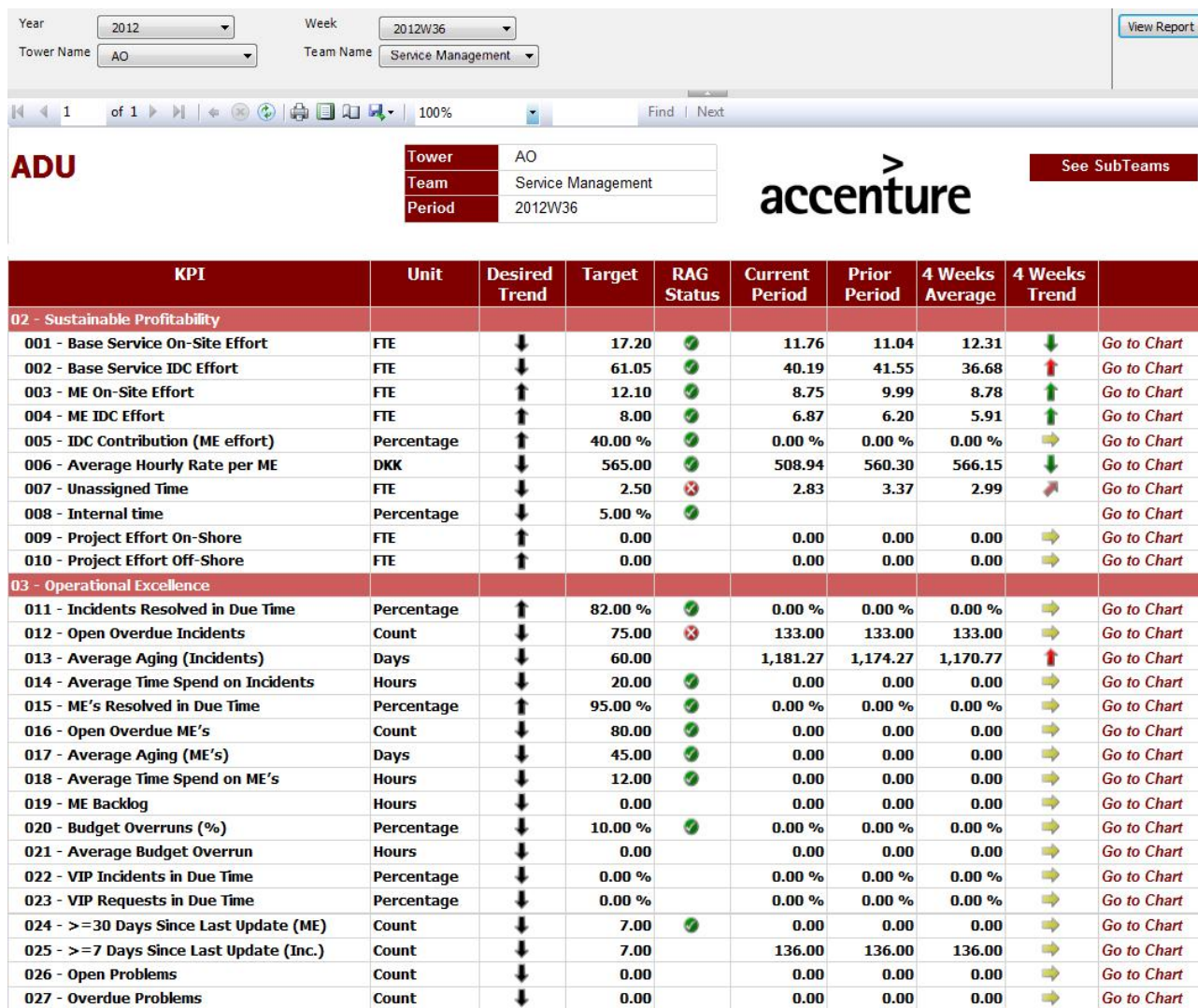


Figura 7.7: Accenture ADU Dashboard di livello Team, realizzato tramite SSRS.

CONCLUSIONI

Questo lavoro di tesi, svolto in ambito aziendale, ha avuto come oggetto la completa automazione della reportistica legata al sistema informativo alla base della collaborazione che la multinazionale Accenture svolge con Velux, una compagnia danese di edilizia, tramite la creazione di un *data warehouse*. In precedenza, a causa dell'inserimento manuale di una consistente quantità di dati e della creazione manuale di alcuni report cruciali come quello relativo agli indicatori di performance, il sistema presentava spesso incongruenze ed errori e le analisi decisionali, di conseguenza, venivano influenzate negativamente. La necessità di automatizzare i processi di estrazione, trasformazione, gestione dei dati e di reportistica tramite la creazione di un *data warehouse* sorge proprio dall'esigenza di porre rimedio a queste inefficienze.

La realizzazione di un solido *data warehouse* permette infatti di assicurare qualità, omogeneità e accuratezza dei dati e di fornire diversi strumenti di analisi e reporting al fine di produrre un'informazione più ricca che porti valore aggiunto all'impresa in termini di miglioramento della capacità decisionale.

Il lavoro è stato lungo e impegnativo a causa di diversi fattori quali le difficoltà incontrate in merito alle rigide regole riguardanti la sicurezza e l'accessibilità dei dati vigenti in Accenture, le alte aspettative ed esigenze in termini di qualità ed efficienza da parte del committente e la dislocazione logistica delle postazioni coinvolte nel processo informativo. Nonostante le difficoltà incontrate, Accenture si è detta molto soddisfatta del prodotto realizzato in quanto, di fatto, sono stati raggiunti tutti gli obiettivi prefissati in fase di riunione preliminare. In particolare, l'utilizzo dello strumento Master Data Services per la gestione dei dati ha suscitato nel committente particolare entusiasmo dal momento che gli strumenti di cui dispone permettono di visualizzare,

mappare e validare i dati e sono strutturati in modo tale da ridurre al minimo le possibili inconsistenze ed errori.

Dopo le fasi preliminari di analisi dei requisiti, progettazione concettuale e logica e ETL, affrontate con attenzione e scrupolosità, l'implementazione del *data warehouse* è stata realizzata grazie al software commerciale già utilizzato dall'impresa. L'utilizzo di un ambiente proprietario ha offerto affidabilità e qualità e ha permesso di rendere molto più veloce lo sviluppo delle procedure di ETL e la creazione dei report.

Le prestazioni del *data warehouse*, inizialmente poco soddisfacenti, sono state migliorate nel corso del progetto. Complessivamente il *data warehouse* non presenta particolari punti critici, tuttavia resta irrisolta la questione riguardante l'interfacciamento con il server Oracle. È proprio questo l'aspetto da curare nell'immediato futuro, sebbene esso sia stato, ovviamente, preso in considerazione nel corso delle varie fasi di progettazione e realizzazione del sistema, il quale è stato progettato in modo tale che l'aggiunta della connessione diretta con il server non vada a influenzare il corretto funzionamento e le performance del *data warehouse*.

Per concludere, l'esperienza lavorativa è stata particolarmente interessante e gratificante sia perché mi ha offerto la possibilità di confrontarmi con la realtà aziendale di un altro paese e di conoscerne processi e meccanismi, sia perché ho potuto migliorare la conoscenza in materia di progettazione e sviluppo di un *data warehouse* e comprendere quali difficoltà e problematiche è necessario affrontare nel caso in cui il committente risulti essere una rinomata multinazionale come Accenture.

BIBLIOGRAFIA

- [Albano 12] A. Albano. Basi di dati di supporto alle decisioni. Università di Pisa, 2012.
- [Ruggieri 12] S. Ruggieri. Analisi dei processi aziendali. Università di Pisa, 2012.
- [Msdn 13] Sito ufficiale Microsoft MSDN,
<<http://www.msdn.microsoft.com/>>, Giugno 2013.
- [Accenture 13] Sito ufficiale Accenture,
<<http://www.accenture.com/>>, Giugno 2013.
- [Velux 13] Sito ufficiale Velux,
<<http://www.velux.com/>>, Giugno 2013.

A mio padre.

*“Ho scelto tutto quello che volevo fare
e ho pagato ben contento di pagare
perché la scelta in fondo è l'unica cosa
che rende questa vita almeno dignitosa.”*